



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

충돌해결과 유저선택을 이용한 Wi-Fi  
무선망에서의 MIMO 성능 향상 기법

**Improving MIMO Performance in Wi-Fi Networks  
by using Collision Resolution and User Selection**

2015년 8월

서울대학교  
전기·컴퓨터공학부  
이 규 행

충돌해결과 유저선택을 이용한 Wi-Fi  
무선망에서의 MIMO 성능 향상 기법

**Improving MIMO Performance in Wi-Fi Networks  
by using Collision Resolution and User Selection**

지도교수 김 종 권

이 논문을 공학박사 학위논문으로 제출함

2015년 8월

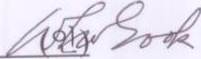

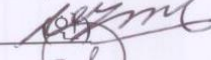
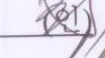

서울대학교 대학원

전기·컴퓨터공학부

이 규 행

이규행의 박사 학위논문을 인준함

2015년 8월

위원장	전 화 숙	
부위원장	김 종 권	
위원	박 세 응	
위원	권 태 경	
위원	유 준	

## ABSTRACT

# Improving MIMO Performance in Wi-Fi Networks by using Collision Resolution and User Selection

Kyu-haeng Lee

Department of Electrical Engineering & Computer Science

The Graduate School

Seoul National University

Multiple-Input Multiple-Output (MIMO) technologies have emerged as a key component to increase the capacity of wireless networks. The MIMO scheme either simultaneously transmits to multiple users at a time or focuses energy towards a single user to enhance the data rate. A number of Wi-Fi standards based on MIMO technology have been developed, and recently, several commercial products have been successfully deployed on the market. Unfortunately, many commercial MIMO-based Wi-Fi products fail to fully exploit the advantages of the MIMO technology, even though the MIMO technology could play a key role in improving the wireless network performance. MIMO nodes cannot provide their higher data rates, especially when they coexist with SISO nodes. Meanwhile, in Wi-Fi networks, significant Channel State Information (CSI) feedback overhead has been obstacle to the performance of MU-MIMO transmission and user selection. Most of these problems are observed to root in the inefficient PHY and MAC design of current MIMO based Wi-Fi systems: the MAC simply abstracts the advancement of PHY technologies as a change of data rate. Hence, the benefit of new PHY technologies are either not fully exploited, or they even may harm the performance of existing network protocols.

In this dissertation we introduce three co-designs of PHY/MAC layers for MIMO based Wi-Fi networks, in order to overcome the intrinsic limitations of the current MIMO based Wi-Fi network and improve the network capacity. First, we show the Interference Alignment and Cancellation (IAC) based collision resolution scheme for heterogeneous MIMO based Wi-Fi systems. Second, we present a practical user selection scheme for MU-MIMO Wi-Fi networks. Finally, we improve the proposed user selection scheme by exploiting a frequency domain signaling scheme and using a capacity gain as a selection metric. We have validated the feasibility and performance of our designs using extensive analysis, simulation and USRP testbed implementation.

**Keywords: MU-MIMO, MAC, Collision, Channel State Information, User Selection**

**Student Number: 2009-23130**

# CONTENTS

<b>ABSTRACT</b> .....	i
<b>CONTENTS</b> .....	iii
<b>LIST OF FIGURES</b> .....	vi
<b>LIST OF TABLES</b> .....	ix
<b>CHAPTER I: Introduction</b> .....	1
1.1 Background and Motivation .....	1
1.2 Goal and Contribution .....	8
1.3 Thesis Organization .....	9
<b>CHAPTER II: MIMO based Collision Resolution</b> .....	10
2.1 Introduction .....	10
2.2 Related Work .....	12
2.3 Background.....	14
2.3.1 Packet Collision Problems in MIMO Networks .....	14
2.3.2 IAC .....	15
2.4 802.11mc .....	17
2.4.1 Protocol Overview .....	17
2.4.2 Packet Collision Resolution via IAC .....	19
2.4.3 Collisions between Multiple CTSs .....	22
2.4.4 Optimal $p$ .....	23
2.4.5 Discussion.....	28
2.5 USRP Experiments .....	33
2.5.1 Micro Benchmark .....	33
2.5.2 Macro Benchmark.....	39
2.6 NS-2 Simulations.....	43

2.6.1 Setting .....	43
2.6.2 Packet Loss Rate due to Collision .....	44
2.6.3 CWMin .....	45
2.6.4 Data Size .....	46
2.6.5 Number of Node Pairs ( $N$ ).....	49
2.6.6 Proportion of MIMO Receivers ( $q_2$ ).....	50
2.6.7 Postamble Probability ( $p$ ) .....	52
2.6.8 Performance in Dynamic Network Configurations.....	54
2.7 Conclusion .....	55
<b>CHAPTER III: User Selection for MU-MIMO Transmission .....</b>	<b>56</b>
3.1 Introduction .....	56
3.2 Related Work .....	58
3.3 Background.....	60
3.3.1 System Model .....	60
3.3.2 User Selection .....	61
3.4 802.11ac+.....	62
3.4.1 Overview.....	62
3.4.2 Channel Hint Broadcasting .....	63
3.4.3 Active CSI Feedback .....	66
3.5 Fair Scheduling.....	72
3.5.1 RR-11ac+ .....	72
3.5.2 PF-11ac+.....	73
3.5.3 Summary.....	73
3.6 Performance Evaluation.....	75
3.6.1 Setting .....	75
3.6.2 802.11ac+ Performance .....	76
3.6.3 Fair Scheduling Protocol Performance .....	79

3.7 Conclusion .....	82
<b>CHAPTER IV: Distributed Frequency Domain User Selection .....</b>	<b>83</b>
4.1 Introduction .....	83
4.2 Motivation .....	84
4.3 DiFuse.....	88
4.3.1 Protocol Overview .....	88
4.3.2 Distributed Feedback Contention .....	89
4.3.3 Slot Threshold Design.....	95
4.3.4 Proportional Fair Selection .....	97
4.3.5 Discussions .....	98
4.4 Performance Evaluation.....	101
4.4.1 Micro Benchmark .....	101
4.4.2 System-Level Performance.....	105
4.5 Conclusion .....	113
<b>CHAPTER V: Conclusion.....</b>	<b>114</b>
<b>BIBLIOGRAPHY.....</b>	<b>115</b>
초 록 .....	122



# LIST OF FIGURES

Figure 1.1: MU-MIMO transmission in 802.11ac. ....	3
Figure 1.2: Throughput gain of user scheduling over 802.11ac.....	5
Figure 2.1: Example of a heterogeneous MIMO network. ....	15
Figure 2.2: Comparison of three schemes: 802.11n, 802.11n+, and 802.11mc. ....	17
Figure 2.3: Example of an mc-RTS. ....	19
Figure 2.4: Generalization to an $M$ -antenna MIMO. ....	21
Figure 2.5: Optimal $p$ when $N = 100$ and $M = 3$ . ....	27
Figure 2.6: Throughput fairness.....	31
Figure 2.7: 802.11mc test environment. ....	33
Figure 2.8: Example of the misalignment.....	35
Figure 2.9: Misalignment in two collided frames. ....	35
Figure 2.10: Success ratio for varying the number of training symbols in the postamble. ....	36
Figure 2.11: SINR enhancement of postamble decoding.....	37
Figure 2.12: Decoding success ratio of two schemes. ....	38
Figure 2.13: Throughput comparison in a macro benchmark. ....	41
Figure 2.14: Packet loss rate due to collision. ....	44
Figure 2.15: Impact of CWMin. ....	45
Figure 2.16: Throughput versus data size with various congestion levels. ....	46
Figure 2.17: Throughput according to the number of pairs.. ....	48
Figure 2.18: Throughput gain of 802.11mc and 802.11n+ over 802.11n.....	48
Figure 2.19: Throughput gain of 802.11mc over other two protocols according to the proportion of MIMO receivers ( $q_2$ ). ....	50
Figure 2.20: Throughput gain of 802.11mc over other two protocols according to the postamble probability ( $p$ ). ....	51
Figure 2.21: $f(p)$ under different network configurations. ....	51

Figure 2.22: System throughput comparison under different network configurations .....	53
Figure 2.23: System throughput comparison under dynamic network configurations. ....	53
Figure 3.1: MU-MIMO downlink system with the $M$ -antenna AP and $K$ single-antenna user stations.....	60
Figure 3.2: Operation example of 802.11ac+.....	62
Figure 3.3: The change in use of the polling frame. ....	64
Figure 3.4: Example of the active CSI feedback with $\gamma$ of five slots.. ....	66
Figure 3.5: Performance of the active CSI feedback. ....	70
Figure 3.6: Throughput according to the number of users.....	76
Figure 3.7: Throughput according to the number of AP antennas. ....	77
Figure 3.8: System throughput according to the number of polls.....	78
Figure 3.9: System throughput comparison of 802.11ac, 802.11ac+ and two fair scheduling protocols.....	79
Figure 3.10: Jain's Fairness Index according to $M$ . ....	80
Figure 3.11: Downlink throughput comparison of 802.11ac, 802.11ac+, RR-11ac+ and PF-11ac+.....	81
Figure 4.1: Normalized capacity gain of each selection metric.....	85
Figure 4.2: An illustrative example of the projected norm based scheme when $M = 2$ and $K = 3$ .....	85
Figure 4.3: Overview of DiFuse operation. ....	88
Figure 4.4: The main concept of slot thresholds of DiFuse. ....	90
Figure 4.5: Three cases of SREQ transmissions.....	91
Figure 4.6: Capacity gain distribution for $M = 2$ and 4.....	94
Figure 4.7: Examples of slot thresholds for different mapping configurations.....	94
Figure 4.8: The gain distribution of the proportional fair utility for $M = 4$ . ....	97
Figure 4.9: Throughput comparison of 802.11ac, SUS, OPUS and DiFuse under hidden terminal environment.....	100
Figure 4.10: Experiment environment. ....	101
Figure 4.11: FFT results under two different transmit power cases.....	102

Figure 4.12: Detection error probability vs. SNR.....	103
Figure 4.13: Performance comparison on the frequency domain contention.....	104
Figure 4.14: Sum-capacity vs. $M$ .....	106
Figure 4.15: CSI feedback duration vs. $K$ .....	108
Figure 4.16: Throughput vs. $M$ .....	109
Figure 4.17: Throughput vs. $K$ .....	111
Figure 4.18: Throughput comparison of 802.11ac, SUS, OPUS-PF and DiFuse-PF.....	112

## LIST OF TABLES

Table 1.1: Summary of the Contributions and Approaches in PHY/MAC Co-Design. ....	7
Table 2.1: Collision Probabilities (in percentage).....	27
Table 2.2: Overhead of Control Frames (in percentage).....	29
Table 2.3: NS-2 Simulation Parameters.....	43
Table 3.1: An Example of the Thresholds.....	66
Table 3.2: Comparison between Three Schemes .....	73
Table 3.3: Default Simulation Parameters .....	75
Table 4.1: Summary of User Selection Protocols .....	98

# CHAPTER I

## Introduction

### 1.1 Background and Motivation

Multiple-Input Multiple-Output (MIMO) technologies have emerged as a key component to increase the capacity of wireless networks. The MIMO scheme either simultaneously transmits to multiple users at a time (i.e., MU-MIMO: Multi-User MIMO) or focuses energy towards a single user to enhance the data rate (i.e., transmit beamforming). A number of Wi-Fi standards based on MIMO technology [1], [2] as well as cellular technologies such as LTE systems [3] have been developed, and recently, several commercial products have been successfully deployed on the market.

Unfortunately, many commercial MIMO-based Wi-Fi products fail to fully exploit the advantages of the MIMO technology, even though the MIMO technology could play a key role in improving the wireless network performance. MIMO nodes cannot provide their higher data rates, especially when they coexist with SISO nodes. Meanwhile, in Wi-Fi networks, significant Channel State Information (CSI) feedback overhead has been obstacle to the performance of MU-MIMO transmission and user selection. Most of these problems are observed to root in the inefficient PHY and

MAC design: the MAC simply abstracts the advancement of PHY technologies as a change of data rate. Hence, the benefit of new PHY (i.e., MIMO) technologies are either not fully exploited, or they even may harm the performance of existing network protocols.

In this dissertation, we have identified the following limitations of MIMO based Wi-Fi networks.

- **Coexistence of heterogeneous MIMO networks**

As a result of the rapid technology advances, there exist heterogeneous nodes with diverse number of antennas in the same network; large devices such as Access Points (APs) or laptops are increasingly equipped with multiple antennas while small devices such as smartphones or sensors have a single antenna due to the limited physical size, capabilities, and cost.

MIMO-based Wi-Fi networks experience the same performance anomaly problem [4] observed in Single-Input Single-Output (SISO)-based Wi-Fi networks; slow nodes consume more channel time than fast nodes under the equal channel access policy, and the overall system throughput decreases. In heterogeneous networks, the performance anomaly phenomenon can be severe because the performance disparity between a SISO node in a poor channel condition and a MIMO node in a good channel condition is large.

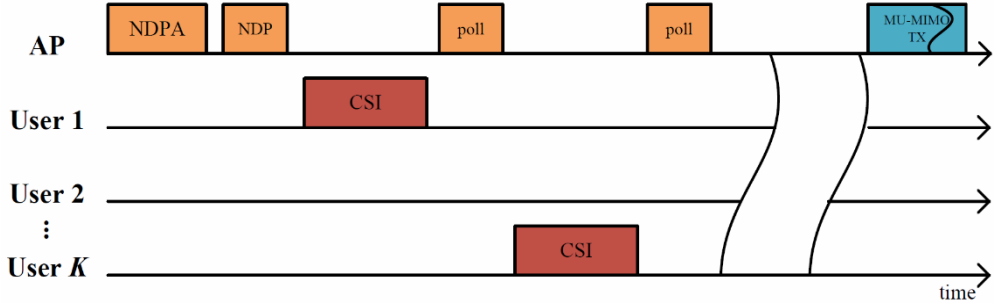


Figure 1.1: **MU-MIMO transmission in 802.11ac.** The AP randomly polls a set of users for MU-MIMO transmission.

- **Significant CSI feedback overhead for MU-MIMO transmission**

To enjoy the high transmission rate of MIMO via interference cancellation or user selection, a transmitter should acquire the CSI from the transmitter to a receiver. To obtain CSI at MIMO transmitters, current technologies use the Request-to-Send/Clear-to-Send (RTS/CTS)-like feedback mechanism [1], [5] or poll-based feedback mechanism [2], as shown in Figure 1.1. However, the CSI feedback overhead can be quite large because those control frames are generally transmitted at the low basic data rate, thus taking a significant portion of the channel time. The CSI feedback overhead can reach up to 25x compared to the data transmission time in the case of 160MHz of bandwidth and 4x1 MIMO [6].

- **Limited scalability for user selection in MU-MIMO networks**

Optimal user selection is essential for increasing the capacity of MU-MIMO Wi-Fi networks. However, determining an optimal user set is difficult and impractical since it requires an exhaustive search over all possible user and antenna sets, and its search space is  $\sum_{m=1}^M \binom{K}{m}$ , where  $K$  and  $M$  are the number of users and the number of AP antennas, respectively. Many researchers have developed greedy user selection algorithms aimed to provide sub-optimal performance while reducing the feedback overhead as well as computational burden [7-12]. The main idea behind the most prior schemes is to incrementally select a user in each iteration by some selection criteria instead of conducting exhaustive search for all user and antenna set

combinations. For example, one user is selected in each iteration such that the new user minimizes interference to previously selected streams.

To accomplish the benefit of the MU-MIMO user selection in the aforementioned algorithms, we need to exploit the multi-user diversity gain; at a given time, the AP can select the best user (e.g., a user with favorable channel conditions) among candidates to improve the system throughput. To leverage multi-user diversity, two key challenges should be addressed: reducing the CSI feedback overhead and employing the proper scheduling policy for user selection.



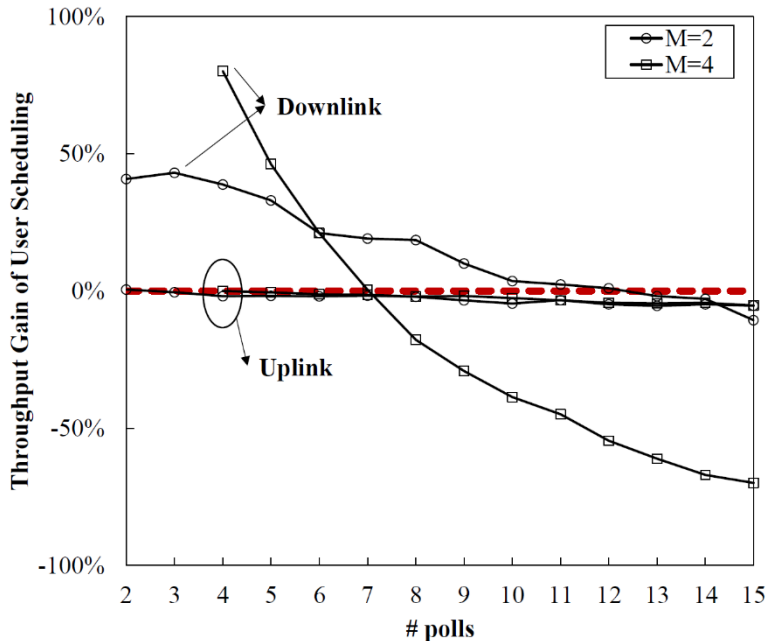


Figure 1.2: **Throughput gain of user scheduling over 802.11ac.**  $M$  and  $K$  stand for the number of AP antennas and the number of users, respectively. In this simulation, we set  $K = 15$ , and all users are assumed to have the same average SNR of 16.7dB. We set 5Mbps and 500kbps of traffic sending rates for downlink and uplink, respectively. As a user scheduling algorithm, we choose Semiorthogonal User Selection (SUS) [9], where a multi-antenna node selects a user if it has the highest ECG (Effective Channel Gain) among users in each selecting round. The red horizontal dashed line indicates the zero gain.

First, the downlink CSI of the candidate users must be efficiently fed back to the AP. Different from cellular systems [3] where separate control channels are used to report the CSI, current 802.11ac Wi-Fi systems use a series of poll-based CSI feedbacks for each user [2], as shown in Figure 1.1. As mentioned before, the CSI feedback overhead can reach up to 25x compared to the data transmission time in the worst case, and such excessive overhead could easily overwhelm the multi-user diversity gain even under optimal user selection. Figure 1.2 shows the throughput gains of a user selection scheme over 802.11ac as a function of the number of polls. As shown in the figure, throughput gain in the downlink decreases with the number of polls. In particular, when the AP accesses the CSI of all users, the loss increases to about 70%. Significant CSI overhead in the downlink also brings a slight throughput loss in the uplink, thereby degrading the whole system performance.

Second, it is vital to select the best user in every user selection step to leverage the MU-MIMO capability by employing the appropriate user selection metric. The projected norm, which is defined as the norm of the user channel projected to the orthogonal subspace of the previously selected user channels [9], is widely used, since it considers both the channel power gain and the orthogonality. However, in some cases, the projected norm based scheme may result in undesirable user selection, due to the fact that it does not consider how the newly joined user channel impacts the already selected ones, if there are any. This may fail to maximize the sum-capacity in each iteration, and occasionally cannot guarantee a positive increment in the sum-capacity. To handle this issue, the AP that employs the norm-based scheme must additionally compute the sum-capacity to assure that it gives positive increment. Here, the feedback report may have induced unnecessary overhead, since the user may not be selected.

Most of the above problems are caused by the way how the MAC layer interfaces with the PHY layer through abstraction. Although MIMO PHY layer features are continuously evolving, the MAC protocols still remain intact, and thus they miss many opportunities to improve the network performance.

Table 1.1: Summary of the Contributions and Approaches in PHY/MAC Co-Design.

Chapter	PHY	MAC	Goal
II (802.11mc)	Postamble and IAC based collision resolution	Random access protocol for collision resolution and concurrent transmission	MIMO based collision resolution scheme for heterogeneous MIMO networks
	<p><i>Contributions:</i></p> <ol style="list-style-type: none"> <li>1. Propose the design of a new MU-MIMO MAC protocol that solves the frame collision problem via postamble.</li> <li>2. Provide a mathematical analysis to derive the optimal probability to add postamble.</li> <li>3. Show the feasibility of 802.11mc by implementing the prototype on USRP/GNURadio and also on NS-2 simulator.</li> </ol>		
III (802.11ac+)	Effective channel gain	Time-domain contention for user selection	Scalable user selection scheme for MU-MIMO Wi-Fi networks
	<p><i>Contributions:</i></p> <ol style="list-style-type: none"> <li>1. Propose the design of an MU-MIMO MAC protocol that accomplishes user selection scheme with a very small amount of CSI feedback.</li> <li>2. Provide a channel hint broadcasting mechanism and an active CSI feedback scheme and two fair scheduling protocols based on 802.11ac+.</li> <li>3. Evaluate the performance of 802.11ac+ via MATLAB simulations.</li> </ol>		
IV (DiFuse)	Overhearing CSI feedbacks and frequency domain signaling	Distributed user selection metric computation and frequency-domain contention	Enhanced user selection protocol for MU-MIMO Wi-Fi networks
	<p><i>Contributions:</i></p> <ol style="list-style-type: none"> <li>1. Propose the design of an enhanced user selection protocol for MU-MIMO Wi-Fi networks.</li> <li>2. Propose a frequency domain contention for user selection and provide a new selection metric.</li> <li>3. Show the feasibility of DiFuse by implementing the prototype on USRP/GNURadio and also on MATLAB simulator.</li> </ol>		

## 1.2 Goal and Contribution

This dissertation proposes three co-designs of PHY/MAC layers for MIMO based Wi-Fi networks, in order to overcome the above limitations and also improve the network capacity. We summarize the contributions and approaches in our PHY/MAC co-designs in Table 1.1.

First, we present a novel MU-MIMO MAC protocol for heterogeneous MIMO based Wi-Fi systems called 802.11 MIMO-based collision resolution (802.11mc for short). The 802.11mc protocol resolves RTS frame collision and extracts CSI from the resolved RTS frames for simultaneous data transmissions. In resolving collided RTS frames, 802.11mc uses the interference handling capability of the multi-antenna array. Then, the receiver sends the CTS reply back to the RTS sender such that it can continue its data frame transmissions as if there were no collisions. In contrast, in the legacy system, the collisions lead to exponential backoff and delayed retransmissions.

Second, we present a practical user selection scheme for MU-MIMO Wi-Fi networks, called 802.11ac+. In 802.11ac+, the AP broadcasts channel information about previously scheduled users by appending it to a poll frame. Then users calculate their effective channel gains from the received information, and the user with the largest gain actively sends a CSI report back to the AP. Upon receiving the CSI report, the AP includes the user in the multi-user transmission schedule and repeats this process while its Degree of Freedom (DoF) constraint is satisfied.

Lastly, we present an enhanced version of our user selection protocol called DiFuse (Distributed Frequency domain user selection). In this scheme, the capacity gain is used as a scheduling metric in user selection: given a user set, the capacity gain of a new user is defined as the increment in network capacity achieved by including the new user to the user set. In particular, each user cleverly computes its expected sum-capacity gain by overhearing the CSI feedback transmissions from other users. Then each user sends its sum-capacity gain in a simplified format via frequency domain contention. The AP uses them to select the best user, then polls it for the actual CSI transmission.

## **1.3 Thesis Organization**

The remainder of this dissertation is organized as follows. In Chapter II we introduce an MU-MIMO MAC protocol for heterogeneous MIMO-based Wi-Fi systems, called 802.11mc. In Chapter III we describe 802.11ac+, the user scheduling protocol for MU-MIMO Wi-Fi networks. In Chapter IV we introduce DiFuse, an enhanced user selection protocol. Finally, Chapter V summarizes the contribution of this dissertation.

# CHAPTER II

## MIMO based Collision Resolution

### 2.1 Introduction

In this chapter, we present a novel MU-MIMO MAC protocol for heterogeneous MIMO-based Wi-Fi systems called 802.11 MIMO-based collision resolution (802.11mc for short). The 802.11mc protocol resolves RTS frame collision and extracts CSI from the resolved RTS frames for simultaneous data transmissions. In resolving collided RTS frames, 802.11mc uses the interference handling capability of the multi-antenna array. Then, the receiver sends the CTS reply back to the RTS sender such that it can continue its data frame transmissions as if there were no collisions. In contrast, in the legacy system, the collisions lead to exponential backoff and delayed retransmissions.

The design of 802.11mc is challenging for the following reasons. When a packet collision occurs, a receiver may decode the overlapping frames by employing Interference Alignment and Cancellation (IAC). To enable IAC, the receiver should obtain at least one clear CSI, which usually can be retrieved on the fly from known

bit sequences such as a preamble. Unfortunately, the medium access mechanism of the 802.11 almost synchronizes the nodes so that the preambles mostly overlap. To solve this problem, we adopt the postamble design at the end of an RTS frame. We call an RTS frame with a postamble an mc-RTS. An mc-RTS provides the CSI when it collides with non-postamble RTS frames (the legacy RTS frames). Using this CSI, the MIMO receiver can efficiently perform the IAC to decode the collided RTS. Furthermore, to handle collisions between more than two mc-RTS frames, the sending nodes add different numbers of postambles according to their receiver types. More specifically, a sender adds  $m-1$  postambles if the target node has  $m$  antennas. Let mc-RTS <sub>$m$</sub>  be an mc-RTS with  $m-1$  postambles. We need to add postambles probabilistically to avoid collisions between mc-RTS frames with the same number of postambles. We take a probabilistic approach and propose an analytic framework to derive the optimal  $p$ , which is the probability of transmitting an mc-RTS.

To show the feasibility of our approach, we implement the 802.11mc prototype on the Universal Software Radio Peripheral (USRP) N210 and GNURadio [13]. We evaluate the performance of 802.11mc via both the USRP testbed experiments and extensive NS-2 simulations. The results show that 802.11mc obtains higher throughput gain than 802.11n and 802.11n+ by addressing the frame collision problem even in the heavily congested network.

We summarize our main contributions as follows. First, we propose the design of a new MU-MIMO MAC protocol, called 802.11mc, that solves the frame collision problem via postamble. Second, we provide a mathematical analysis to derive the optimal probability to add postamble. Finally, we show the feasibility of 802.11mc by implementing the prototype on USRP/GNURadio and also on NS-2 simulator.

The remainder of this chapter is organized as follows. Section 2.2 provides the related work, and we give our motivation and a description of the IAC in Section 2.3. We next describe the 802.11mc mechanism in a greater detail in Section 2.4. Section 2.5 shows the performance evaluation based on USRP experiments, and Section 2.6 provides NS-2 simulation results. We finally conclude this chapter in Section 2.7.

## 2.2 Related Work

We survey the research results on the collision resolution and concurrent transmission techniques related to 802.11mc.

### **Collision Handling Schemes**

Recent advances in signal processing enable overcoming the problem in conventional wireless transmissions, i.e., discard all collided frames. Partial recovery schemes utilize known symbols to recover collided frames and reduce the frame retransmissions [14], [15]. ZigZag decoding [16], Chorus [17], and analog network coding [18] use some powerful physical layer techniques such as successive interference cancelation and analog network coding. These schemes, however, are limited only to very specific collision cases, e.g., successive collisions by hidden terminals.

Similar to the collision detection capability in wired networks, several methods [19], [20] that notify collisions to transmitters in wireless networks have been proposed. Carrier Sense Multiple Access with Collision Notification (CSMA/CN) [20] enables transmitting nodes to detect a collision during transmission and force them to abort transmission. In order to realize this scheme, transmitters have to use the full-duplex function of a multi-antenna array. This assumption does not fit to our network environment where common SISO and MIMO nodes coexist in the network.

### **MU-MIMO Concurrent Transmission**

In MU-MIMO protocols, multiple users can transmit and receive independent data streams simultaneously. MU-MIMO transmissions can occur both in the infrastructure mode and the ad hoc mode. Let us consider first downlink transmission in the infrastructure. In this case, MIMO-equipped APs can transmit different data to multiple users simultaneously by using precoding. Precoding requires *a priori* apprehension of channel information, which incurs significant channel feedback overhead [6]. This is the main reason why downlink MU-MIMO feature is not included as mandatory but is an option in an IEEE 802.11ac standard [2]. Concurrent upstream transmissions to an AP in the infrastructure mode are even more difficult than downstream communications because client nodes synchronize their



transmissions to realize beamforming. Instead, staggered transmission methods can be used so that the MIMO AP or base stations can decode overlapped uplink frames one by one [21], [22]. In the ad hoc mode, beamforming is required on both senders and receivers. IAC [23] connects multiple APs to each other with a backbone wired network such that connected APs act as a single virtual MIMO AP. IAC achieves a higher degree of concurrence than other MIMO schemes. However, to handle interference, APs must somehow exchange the decoded packet, e.g., using wired transmission.

A novel MU-MIMO MAC called 802.11n+ [5] enables distributed random accesses. In 802.11n+, a MIMO node can send its frame along with the ongoing transmissions from other SISO nodes if there are unused DoFs. To realize this, the authors propose two schemes, i.e., carrier sensing in a multi-dimensional space and interference nulling and alignment. First, 802.11n+ nodes need to carrier sense in the presence of ongoing transmissions. To do this, they project the received signal on a space orthogonal to the ongoing transmissions in a multi-dimensional space that a multi-antenna array creates. Second, for MIMO nodes to transmit without interfering the ongoing transmissions, they should perform the interference nulling and alignment. This scheme, however, requires MIMO transmitters to know the CSI from the receivers of ongoing transmissions, as well as their own receivers. To do this, 802.11n+ requires RTS/CTS exchange before data transmission. In this system, the overall system throughput significantly increases because MIMO nodes have additional transmission chances concurrent to ongoing transmissions of SISO nodes. While several ideas of 802.11n+ may bear some resemblance to our scheme, the SISO nodes in 802.11n+ cannot enjoy any additional transmission opportunities. Rather, 802.11n+ requires the RTS/CTS handshake and may even degrade the performance of SISO nodes, although they use a lightweight RTS/CTS [24] to lower the overhead. In contrast, our scheme increases the transmission opportunities, even for the SISO nodes and the MIMO nodes.

## 2.3 Background

### 2.3.1 Packet Collision Problems in MIMO Networks

The packet collision problem has long been highlighted in wireless communication systems because of its adverse effect on wireless capacity. For example, the well-known incident of Steve Jobs' iPhone keynote presentation failure in 2010 due to arduous access contentions shows the severity of this problem. Until now, numerous collision resolution methods have been proposed. The RTS/CTS mechanism, based on a simple intuition, i.e., RTS/CTS frames take much less channel duration compared with data frames, is an attractive solution to ease the burden of such collision problems. In addition, many researchers exploit the RTS/CTS mechanism as a versatile tool to achieve different purposes such as rate adaptation and CSI estimation. Since RTS/CTS frames are transmitted at the basic data rate (e.g., 6.5Mbps in 802.11n), however, RTS/CTS collision time can be quite large in high-data-rate WLAN systems such as 802.11n or 802.11ac. For example, if a retransmission due to a collision of RTS frames occurs and transmits a 1500-byte frame through a channel of 130Mbps data rate, at least two RTS frames, i.e., one CTS frame and one DATA frame transmissions, are required. The channel duration periods for the control and data frames are  $126.46\mu s$  and  $154.15\mu s$ , respectively, which means that about 45% of transmission time is being consumed for the RTS/CTS control exchange when a collision happens.

802.11mc resolves collisions in MIMO networks, may it be an RTS. This would result in a significant system throughput increase. To realize this, 802.11mc exploits the interference-handling capability of a multi-antenna array (e.g., IAC and postamble), whereas most previous proposals fail to fully exploit the capabilities of multi-antenna techniques.

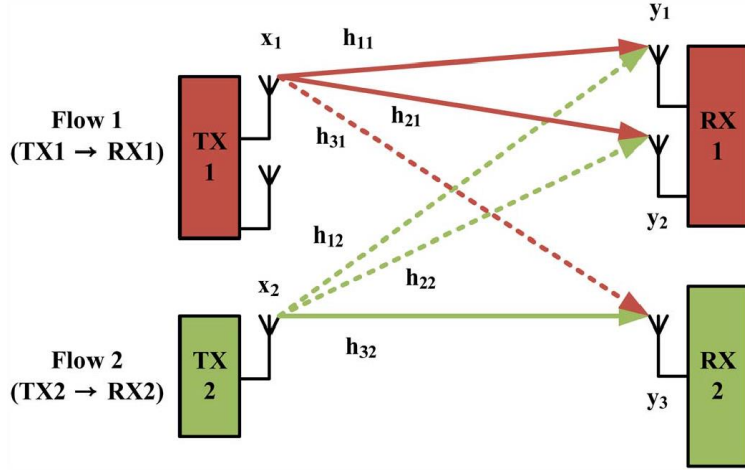


Figure 2.1: **Example of a heterogeneous MIMO network.** Two nodes TX1 and TX2 send frames to their receivers RX1 and RX2, respectively. This topology is a heterogeneous MIMO network, where the first flow is MIMO, and the second flow is SISO.

### 2.3.2 IAC

IAC is a key physical layer technique that enables the MIMO receiver to recover the signal when packet collisions occur. Consider two pairs of nodes, i.e., (TX1, RX1) and (TX2, RX2), in a heterogeneous MU-MIMO system, as shown in Figure 2.1. We assume that the multi-antenna node TX1 uses only one antenna in this example. Transmitters TX1 and TX2 simultaneously send frames, i.e.,  $x_1$  and  $x_2$ , to their receivers RX1 and RX2, respectively. Each receiver receives a combined signal of the two concurrent transmissions, and a collision occurs. Now, we explain how a multi-antenna receiver RX1 can decode all collided frames, whereas a single-antenna receiver RX2 cannot. We denote  $h_{rt}$  as a channel state from the transmitting antenna  $t$  of transmitters to the receiving antenna  $r$  of receivers, e.g.,  $h_{21}$  means the channel state from antenna 1 of TX1 to antenna 2 of RX1 in Figure 2.1. Then, received signals at RX1 are represented as<sup>1</sup>

$$y_1 = h_{11}x_1 + h_{12}x_2 \quad (1)$$

<sup>1</sup> For simplicity, we omit the noise terms.

$$y_2 = h_{21}x_1 + h_{22}x_2. \quad (2)$$

Our purpose is to decode all received frames, i.e.,  $x_1$  and  $x_2$ . If all CSI is given, we can readily decode the frames by solving linear equations (1) and (2), as do typical MIMO decoders [25, [26]. IAC is a new technology that recovers two concurrent frames knowing only one CSI. Let us assume that RX1 already knows the CSI of TX1, i.e.,  $H_1 = \begin{pmatrix} h_{11} \\ h_{21} \end{pmatrix}$ . (We will explain how a multi-antenna receiver can obtain this CSI in Section 2.4.2.)

From (1) and (2), we obtain

$$h_{21}y_1 - h_{11}y_2 = h_{21}h_{12}x_2 - h_{11}h_{22}x_2. \quad (3)$$

Substituting the left-hand side of (3) into  $y'$ , we finally obtain

$$y' = (h_{21}h_{12} - h_{11}h_{22})x_2. \quad (4)$$

Equation (4) represents that the linear combination of received signals is composed of only one signal  $x_2$ . This procedure is IA, which aligns more than two signals along with a target direction to nullify one signal. In the previous example, the alignment nullifies signal  $x_1$  and, hence, enables decoding of signal  $x_2$ . The next step is to decode the other frame  $x_1$ . Fortunately, since we already decoded  $x_2$ , we can reconstruct the original form of  $x_2$ . Canceling  $x_2$  from (1) or (2), we can finally decode the signal  $x_1$ . Note that the whole frame of  $x_2$  can be used as training symbols to estimate the CSI from TX2 [21].

The IAC is easily generalized for  $M$ -antenna MIMO receivers. According to the theory [26], an  $M$ -antenna MIMO receiver can decode maximum  $M$  overlapping signals using the IAC repeatedly. Recently, several researchers have been utilizing this method to their systems [5], [16], [21-23]. We give more details of our protocol in the following section.

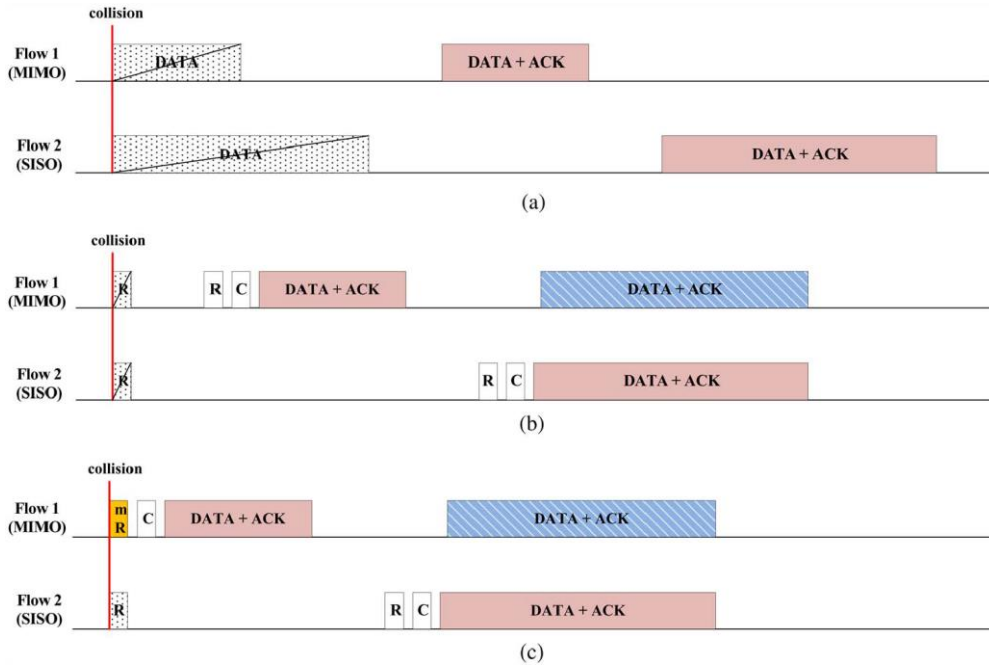


Figure 2.2: **Comparison of three schemes: 802.11n, 802.11n+, and 802.11mc.** 802.11mc provides higher throughput by collision resolution and concurrent transmission. In the figure, letters “R,” “C,” and “mR” represent RTS, CTS, and mc-RTS frames, respectively. A blue-colored box represents the concurrent frame transmission from a MIMO node. (a) 802.11n. (b) 802.11n+. (c) 802.11mc.

## 2.4 802.11mc

### 2.4.1 Protocol Overview

We compare the operation of 802.11mc with those of 802.11n and 802.11n+ in Figure 2.2 under the network topology<sup>2</sup> shown in Figure 2.1. In this example, a frame collision happens at the beginning. Note that we assume that the 802.11n does not use

<sup>2</sup> We consider a heterogeneous MIMO network, where each device communicates with another device, similar to 802.11n+. For example, inside a single house, a user connects her smartphone to a multi-antenna AP, while a home controller with a single antenna communicates with sensor nodes.

RTS/CTS, and 802.11n+ and 802.11mc use RTS/CTS for CSI measurements. Additionally, we assume that a MIMO node may use all antennas in data transmission but use only one antenna in RTS transmission.

Let us first examine the 802.11n operation, as shown in Figure 2.2 (a). Since 802.11n can only operate in the SU-MIMO mode, it cannot resolve the frame collision. In case of RTS frame collision, receivers fail to recover the frames and both transmitters back off to the next round [see Figure 2.2 (b)]. If the MIMO node captures the medium, it uses the medium exclusively. However, if the SISO node captures the medium, the MIMO node can transmit concurrently with the SISO node.

Meanwhile, when RTS frames collide, 802.11mc may not require the RTS retransmissions because MIMO receivers can decode all the collided frames, as shown in Figure 2.2 (c). As a result, 802.11mc gives transmission opportunities even under packet collision, providing better channel utilization than 802.11n or 802.11n+. We propose a postamble frame structure to increase the opportunity to receive clean signals of known bit pattern.

The part of achieving DoF for concurrent transmission in 802.11mc is basically similar with that of 802.11n+. We understand that the method to exploit DoF, e.g., aligning signals from different spatial domain channels into one target spatial domain, is not new and has already been used in other recent MIMO systems. In order to achieve this goal, the CSI should be acquired prior to the transmission, and both 802.11mc and 802.11n+ utilize the RTS/CTS for this purpose. One difference from 802.11n+ is that we can reduce the number of control frame exchanges because 802.11mc MIMO nodes can extract the CSI, even when a collision occurs.

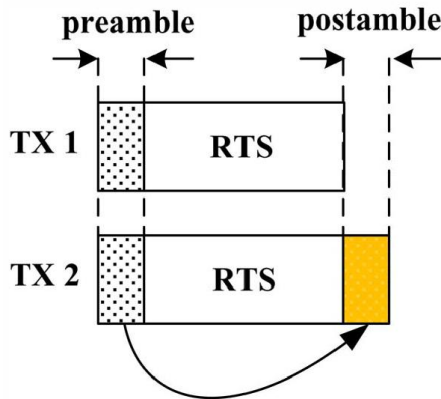


Figure 2.3: **Example of an mc-RTS.** A collision prevents receivers estimating the CSI from preambles. In an mc-RTS, a node duplicates and appends a preamble at the end of the RTS frame.

## 2.4.2 Packet Collision Resolution via IAC

As previously described, we can recover collided frames, applying IAC if one of the channel states is known. A clever way to acquire the CSI is to guess the channel state by comparing arrived preamble signals with expected signals of the same known bit sequence. However, when a packet collision occurs, the collided frames typically overlap from the beginning and the preambles are typically corrupted. One method to protect the preamble is to artificially regulate transmission start times [21], [22]. A node intentionally transmits its packet after the preamble duration of another ongoing transmission so that the two preambles do not overlap. However, this procedure fails to avoid collision completely, e.g., collisions of multiple nodes with the same backoff counters.

If only one of the nodes transmits an mc-RTS, then a MIMO receiver can measure the CSI from the clean postamble signal, as shown in Figure 2.3. If both transmitters use mc-RTSs, then we cannot receive a clean postamble, and thus, IAC fails. To increase the probability of clean postambles, we use two approaches. One is to append postamble probabilistically; not all RTS frames have postambles, but a postamble is appended with probability  $p$ . Another approach is to vary the number of postambles according to receiver antenna configuration. More specifically, a transmitter pads postambles in proportion to the number of its receiver's antennas. Recall that the

number of postambles is *not* determined by the number of antennas of the transmitter but of the receiver. For example, an mc-RTS<sub>*m*</sub> for an *m*-antenna receiver has *m* − 1 postambles. Note that a transmitter appends *m* − 1 postambles with probability *p* if its receiver has *m* antennas.



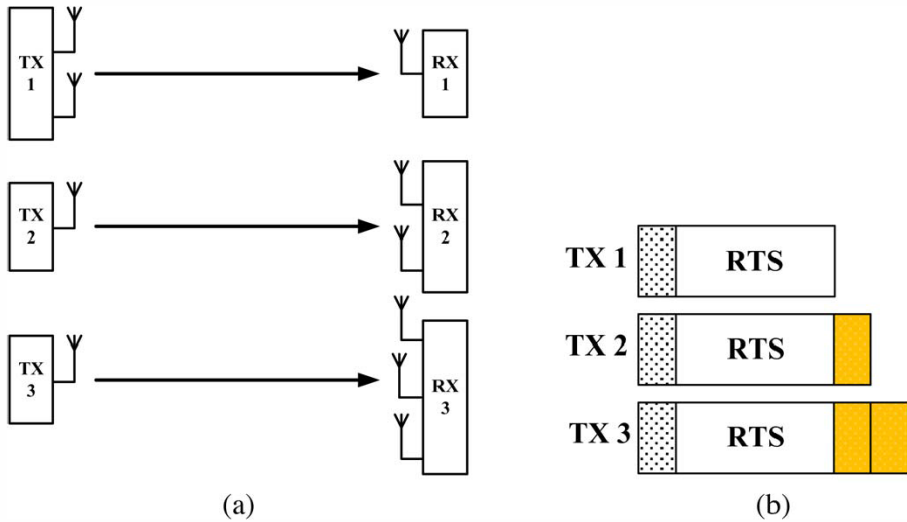


Figure 2.4: **Generalization to an  $M$ -antenna MIMO.** Each transmitter pads multiple postambles in proportion to the number of its receiver’s antennas, as shown in (b). In this example, only RX3 can decode up to three overlapping signals by repeatedly using the IAC. (a) Example of generalized networks. (b) mc-RTS for each receiver.

Opportunistic postambles make 802.11mc handle the various collisions between more than two frames. Let us consider an example with three transceiver pairs in Figure 2.4 (a). In this example, three transmitters simultaneously send RTS frames to their receivers. Recall that we assume that RTS frames are sent by using a single antenna. TX1 cannot append postambles since its receiver RX1 is a single-antenna node. Other transmitters, i.e., TX2 and TX3, use mc-RTS2 and mc-RTS3, respectively. Then, each receiver receives the overlapping frames, as shown in Figure 2.4 (b). RX3 can decode overlapping signals by repeatedly using the IAC and send a CTS back to its transmitter TX3, whereas others cannot decode the signals due to their DoF constraints.

The use of postamble is the only change we have made in this work, and it does not render any issues on the interoperation of legacy and 802.11mc nodes. First, when 802.11mc nodes send a legacy RTS, nodes will work as the legacy 802.11 systems. Second, when an mc-RTS is transmitted without collision, nodes just decode the frame by using the preamble. Third, when an mc-RTS is transmitted with collision, 802.11mc nodes detect the collision and perform decoding from the postamble,

whereas the legacy nodes may regard the postamble as a regular preamble but cannot do anything with it.

The 802.11mc (MIMO) receivers should maintain a buffer of samples of previously received symbols. Fortunately, the length of the RTS is fixed in 802.11 systems, and those of postambles are determined by the number of receiver antennas. Thus, receivers may simply keep the buffer size at least the length of their mc-RTSs.

### 2.4.3 Collisions between Multiple CTSs

Next, consider the case where multiple MIMO receivers succeed in decoding the collided frames using the mc-RTS. Here, a CTS collision can happen since they will attempt to send the CTS simultaneously. To solve the CTS collision, we give the right to send a CTS to only one MIMO receiver whose sender has sent the longest mc-RTS before. This guarantees that only one receiver sends a CTS at a time even if multiple MIMO receivers decode the collided RTS frames. Let us assume that RX2 in Figure 2.1 is equipped with three antennas and TX1 and TX2 send mc-RTS<sub>2</sub> and mc-RTS<sub>3</sub> to their receivers, respectively. In this case, both receivers can decode the collided frames with their enough DoFs and RX1 cannot send a CTS to TX1. Instead, RX2 sends a CTS to TX2, i.e., the length of mc-RTS<sub>2</sub> < the length of mc-RTS<sub>3</sub>.

As shown in the previous example, IAC may decode multiple overlapped mc-RTS frames, but the longest mc-RTS is clearly the one that is decoded last, which is the mc-RTS with the highest priority. Then, only the receiver that matches the last mc-RTS's destination address transmits the CTS, whereas other receivers remain silent. Therefore, the consensus is implicitly achieved. Furthermore, since a transmitter adds the postambles in proportion to the number of its receiver's antennas, the receiver can easily match the length of the postambles with its own number of antennas.

Now, we provide the optimal probability  $p$  in the following section.

### 2.4.4 Optimal $p$

If two mc-RTSs with the same number of postambles collide, then we cannot decode the postamble from the two mc-RTSs since they will overlap. Therefore, we present an optimal postamble appending probability  $p$ , which determines whether it will use the mc-RTS; otherwise, stick with the legacy RTS. This helps the 802.11mc collision resolution to become effective.

Assume that there are  $N$  transceivers. Let  $M$  be the maximum number of receiver antennas and  $N_m$  be the number of  $m$ -antenna receivers.

Let us define function  $f(p)$  as the total number of resolvable collisions in 802.11mc. Then, our objective is to find the optimal  $p^*$  that maximizes  $f(p)$ :

$$p^* = \arg \max_{0 < p \leq 1} f(p). \quad (5)$$

Since  $M$ -antenna receivers can recover collisions involved with up to  $M$  frames, we obtain

$$f(p) = \sum_{m=2}^M f_m(p) \quad (6)$$

where  $f_m(p)$  represents the number of resolvable collisions between  $m$  frames, and it is categorized into the following three cases:

- $f_m^1(p)$ : the number of collisions where one of the collided frames is for a SISO receiver;
- $f_m^2(p)$ : the number of collisions where all collided frames are for MIMO receivers that have a different number of antennas from each other;
- $f_m^3(p)$ : the number of collisions where all collided frames are for MIMO receivers and two of them have the same number of antennas.

Now, for nonnegative integer numbers  $N_1, N_2, \dots, N_M$  and binary numbers  $r_1, r_2, \dots, r_{M-1}$ , each case is represented as follows.

1)  $f_m^1(p)$

Since one of the  $m$  collided frames is for a SISO receiver, it must use a legacy RTS without the postamble. In the same way, the other  $m - 1$  frames must be mc-RTSs

since they are intended for MIMO receivers. To recover the collided frames, collided mc-RTSs should have different numbers of postambles from each other. Thus, we obtain the following:

$$\begin{aligned}
f_m^1(p) &= p^{m-1} \left\{ \binom{N_1}{1} \binom{N_2}{1} \dots \binom{N_m}{1} + \dots + \binom{N_1}{1} \binom{N_{(M-m+2)}}{1} \dots \binom{N_M}{1} \right\} \\
&= \binom{N_1}{1} p^{m-1} \left\{ \binom{N_2}{1} \dots \binom{N_m}{1} + \dots + \binom{N_{(M-m+2)}}{1} \dots \binom{N_M}{1} \right\} \\
&= (N_1) p^{m-1} \sum_{r_1 + \dots + r_{(M-1)} = m-1} (N_2)^{r_1} (N_3)^{r_2} \dots (N_M)^{r_{(M-1)}} \\
&= (N_1) p^{m-1} \sum_{r_1 + \dots + r_{(M-1)} = m-1} \prod_{i=2}^M (N_i)^{r_{(i-1)}} \\
&= (N_1) p^{m-1} A(M, m) \tag{7}
\end{aligned}$$

2)  $f_m^2(p)$

Here, we present the two cases to recover the collided frames: 1) One of the transmitters involved in the collision does not use mc-RTS; and 2) all transmitters involved in the collision use mc-RTSs. For the first case, since all  $m-1$  transmitters use mc-RTSs except one transmitter, the probability is  $mp^{m-1}(1-p)$ . For the second case, the probability that all transmitters use mc-RTSs is  $p^m$ . We obtain the following:

$$\begin{aligned}
f_m^2(p) &= \{p^m + mp^{m-1}(1-p)\} \left\{ \binom{N_2}{1} \binom{N_3}{1} \dots \binom{N_{(m+1)}}{1} + \dots \right. \\
&\quad \left. + \binom{N_{(M-m+1)}}{1} \binom{N_{(M-m+2)}}{1} \dots \binom{N_M}{1} \right\} \\
&= \{p^m + mp^{m-1}(1-p)\} \sum_{r_1 + \dots + r_{(M-1)} = m} (N_2)^{r_1} (N_3)^{r_2} \dots (N_M)^{r_{(M-1)}} \\
&= \{p^m + mp^{m-1}(1-p)\} \sum_{r_1 + \dots + r_{(M-1)} = m} \prod_{i=2}^M (N_i)^{r_{(i-1)}} \\
&= \{p^m + mp^{m-1}(1-p)\} B(M, m). \tag{8}
\end{aligned}$$

3)  $f_m^3(p)$

In this case, to avoid overlapping of two mc-RTSs with the same number of postambles, only one of the transmitters of those frames should use mc-RTS, and the other  $m - 2$  frames should be mc-RTSs. Thus, the probability is  $2p^{m-1}(1 - p)$ , and finally, we represent  $f_m^3(p)$  as

$$\begin{aligned}
f_m^3(p) &= 2p^{m-1}(1 - p) \left\{ \binom{N_2}{2} \binom{N_3}{1} \dots \binom{N_m}{1} + \binom{N_2}{1} \binom{N_3}{2} \dots \binom{N_m}{1} + \dots \right. \\
&\quad \left. + \binom{N_{(M-m+2)}}{1} \binom{N_{(M-m+3)}}{1} \dots \binom{N_M}{2} \right\} \\
&= p^{m-1}(1 - p) \sum_{r_1 + \dots + r_{(M-1)} = m-1} (N_2)^{r_1} \dots (N_M)^{r_{(M-1)}} \\
&\quad \times \{ (N_2)^{r_1} + \dots + (N_M)^{r_{(M-1)}} - (m - 1) \} \\
&= p^{m-1}(1 - p) \sum_{r_1 + \dots + r_{(M-1)} = m-1} \prod_{i=2}^M (N_i)^{r_{(i-1)}} \\
&\quad \times \left\{ \sum_{i=2}^M (N_i)^{r_{(i-1)}} - (m - 1) \right\} \\
&= p^{m-1}(1 - p) C(M, m). \tag{9}
\end{aligned}$$

By plugging (7)–(9) into (6)

$$\begin{aligned}
f(p) &= \sum_{m=2}^M \{ [(1 - mB(M, m) - C(M, m))] p^m \\
&\quad + [(N_1)A(M, m) + mB(M, m) + C(M, m)] p^{m-1} \} \\
&= \{ (1 - M)B(M, M) - C(M, M) \} p^M + \dots \\
&\quad + \{ (N_1 + 1 - M)B(M, m) + MB(M, m + 1) + C(M, m + 1) \\
&\quad - C(M, m) \} p^m + \dots + \{ (N_1)B(M, 1) + 2B(M, 2) + C(M, 2) \} p \\
&= -C(M, M)p^M + \dots + \{ (N_1)B(M, 1) + 2B(M, 2) + C(M, 2) \} p. \tag{10}
\end{aligned}$$

As shown in (10), since  $f(p)$  is an  $M$ -degree function of  $p$ , it is unfeasible to directly obtain the optimal  $p$  to maximize the function. Instead, we use a hill-

climbing method [27]. Note that this method occasionally gets stuck in the local maximum. To mitigate this problem, we select multiple initial points that are evenly spaced and repeat the algorithm starting from these points.

Table 2.1: **Collision Probabilities (in percentage)**

		the number of frames involved in a collision						
		2	3	4	5	6	7	8
$N$	10	19.32	2.85	0.28	0.02	0.00	0.00	0.00
	20	23.56	4.96	0.74	0.08	0.01	0.00	0.00
	30	25.68	6.37	1.14	0.16	0.02	0.00	0.00
	40	27.05	7.46	1.50	0.24	0.03	0.00	0.00
	50	28.03	8.35	1.83	0.31	0.04	0.01	0.00
	100	30.63	11.52	3.22	0.71	0.13	0.02	0.00

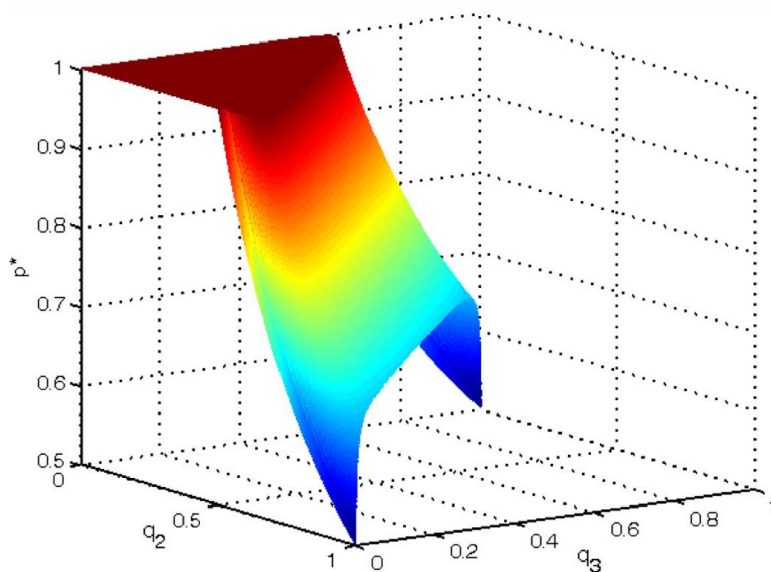


Figure 2.5: **Optimal  $p$  when  $N = 100$  and  $M = 3$ .** Optimal  $p$  is highly related to  $q_m$ . As shown in the graph, the optimal  $p$  spans from 0.5 to 1.

The complexity of this optimization mainly lies on  $M$ . Since  $M$  is no more than four in typical 802.11n-based Wi-Fi systems, it is simple to obtain the coefficients. In the near future, however, wireless devices equipped with many number of antennas, such as massive MIMO systems [28-31], will appear, and they may use more space-time streams than the current devices<sup>3</sup>. Fortunately, the actual probabilities of the

<sup>3</sup> We note that 802.11ac [2] will support up to eight space-time streams.

collision with more than four frames are very small and hence negligible. In Table 2.1, we provide the collision probabilities according to  $N$  and the number of frames involved in a collision with CWMin of 15. Thus, if we consider the limited number of streams, then we can obtain the equations in a much simpler form.

As an example, we provide the optimal  $p$  by varying the proportion of receiver types with  $M = 3$ , i.e.,  $q_1 + q_2 + q_3 = 1$ , where  $q_m$  is the ratio of  $m$ -antenna receivers to the total number of receivers. As shown in Figure 2.5,  $p^*$  stays at 1 until the proportion of MIMO receivers ( $q_2 + q_3$ ) reaches a certain critical point, and after that, it gradually decreases as the proportion increases. In particular, when it reaches 1, if there is only one type of MIMO receiver, the optimal  $p$  becomes 0.5 (i.e., when  $q_2 = 1$  or  $q_3 = 1$ ,  $p^* = 0.5$ ); otherwise, it increases (i.e., when  $q_2 + q_3 = 1$ , maximum  $p^* = 0.67$ ). Intuitively, this happens because the diverse number of MIMO receiver antennas increases the total number of resolvable collisions.

## 2.4.5 Discussion

### 1) *Modification*

We slightly modify the 802.11 standard to employ mc-RTS in 802.11mc; the postamble is appended at the end of the current 802.11 RTS frame. The postamble takes exactly the same format as the preamble, although the location is different. In case the legacy 802.11 or SISO node recognizes the postamble, they would regard this as a preamble and try to decode the following signals in vain. Therefore, this feature is backward compatible with the legacy 802.11 nodes.

### 2) *Overhead*

The length of mc-RTS $_m$  increases with  $m$ , specifically  $m - 1$  training symbols. Typically, 802.11n uses two long training Orthogonal Frequency Division Multiplexing (OFDM) symbols in a single preamble, e.g.,  $4\mu\text{s}$  for each. Thus, even for mc-RTS $_4$ , the additional overhead compared with mc-RTS $_1$  is only  $12\mu\text{s}$ , which is negligible for a typical wireless network.



Table 2.2: **Overhead of Control Frames (in percentage)**

MCS Index	802.11mc			802.11n+
	$M$			-
	2	3	4	
0	4.68	4.86	5.05	4.49
1	8.48	8.80	9.12	8.15
2	11.62	12.04	12.46	11.19
3	14.26	14.76	15.26	13.75
4	18.46	19.08	19.69	17.83
5	21.64	22.34	23.02	20.93
6	22.96	23.69	24.40	22.22
7	24.14	24.90	25.63	23.38

Table 2.2 shows the overhead of the control frames (RTS/CTS) under 802.11mc and 802.11n+ according to different Modulation and Coding Scheme (MCS) indexes. We use a single-stream case with a 1500-byte payload. Recall that 802.11mc increases the size of the RTS by appending postambles. However, the overhead increases only slightly; compared with 802.11n+, it is increased only by up to 2.25% (in the case of  $M = 4$  and MCS index of 7). However, as demonstrated in our results, the gain from using the control frame exchange is more than enough to overcome its overhead for both 802.11n+ and 802.11mc. Furthermore, a lightweight RTS/CTS [24], which reduces the overhead, can be easily employed to the 802.11mc. As will be shown in Figure 2.22, the system throughput is increased by 14% from the lightweight handshake due to the reduced overhead.

### 3) *Different Postamble Probabilities*

802.11mc can use another mc-RTS scheme using different postamble probabilities with the same MC-RTS, i.e.,  $\text{mc-RTS}_2 = \text{mc-RTS}_3 = \dots = \text{mc-RTS}_M$ , according to the MIMO receiver type. In this scheme, the protocol has a less overhead but cannot resolve any collisions between more than two frames. Our opinion is that we can adaptively use different mc-RTS rules by considering network configurations such as the number of nodes and receiver antenna diversity. We leave this issue to our future work.

#### 4) *Rate Control*

802.11mc can easily adopt various RTS/CTS-based rate adaptation algorithms [32], [33]. The rate adaptation method is parallel to the mc-RTS; thus, they do not affect each other.

#### 5) *Estimating $N$ and Receiver Configuration*

To exploit the optimal  $p$ , the transmitters need to estimate the number of stations  $N$ , and the receiver configuration, e.g., the number of receivers' antennas. First, there are numerous algorithms [34], [35] that estimate the number of contending nodes  $N$ , and we can simply adopt one of those schemes. These schemes are known in providing quite accurate and fast estimation results within error of 1% in 1s. Second, since the 802.11n systems are based on the WLAN OFDM system, two new formats are defined for the Physical Layer Convergence Protocol (PLCP) for MIMO nodes: the mixed mode and the green field. In these modes, the MIMO training sequence format is added, and this enables any node to know others' antenna configurations because its length is typically equal or greater than the number of space-time streams. In practical systems, it is sufficient to measure only the number of SISO nodes and that of MIMO nodes (regardless of the number of antennas) because the majority of packet collision happens between the two nodes, as shown in Table 2.1. Note that the postamble probability  $p$  ranges from 0.5 to 1 for any cases. Even if a node picks a non-optimum value, 802.11mc can guarantee the minimum level of performance gain over 802.11n and 802.11n+.

#### 6) *Behavior in Multi-collision Domains*

Until now, we have considered the single-collision-domain scenario, but in the real world, there could be some nodes belonging to multiple collision domains. Such nodes may have limited concurrent transmission opportunities because their DoFs are bounded by the multiple collision domains. For the collision resolution, 802.11mc nodes should be in a transmission range where nodes can listen to the RTS/CTS from each other. A final note is that the nodes in the multiple collision domains may suffer from starvation [36]; hence, we should consider this to redesign the backoff mechanism. We leave further study of the multiple collision domains as future work.

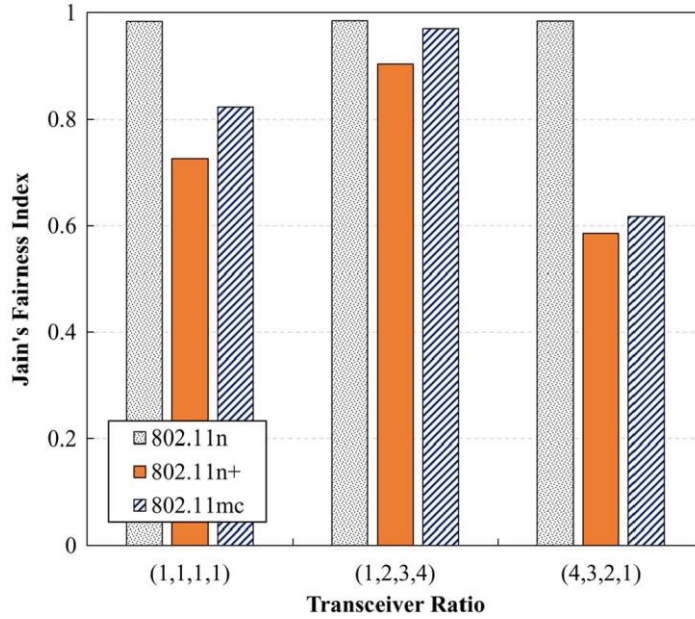


Figure 2.6: **Throughput fairness.** There are 20 transceiver pairs with four types of nodes from one to four antennas. In this figure,  $(:, :, :, :)$  denotes the ratio of SISO, two-antenna MIMO, three-antenna MIMO, and four antenna MIMO pairs, respectively. For example,  $(1,1,1,1)$  means that the network has equal number of pairs for four different types.

### 7) Achieving Fairness

First, 802.11mc MIMO nodes can gain more transmission opportunities by exploiting the concurrent transmission due to DoF. Thus, MIMO nodes with higher DoF achieve higher throughput, and this makes sense in terms of proportional fairness; higher capable nodes, e.g., MIMO nodes, get better performance. In contrast, the random access nature of the current 802.11n gives the same throughput to all heterogeneous nodes, regardless of their capabilities, and this may cause the overall system performance to degrade, i.e., performance anomaly [4]. Another point to make is that nodes with low DoF may have less chance to access the channel after the collision resolution. One way of overcoming this issue is to let all nodes involved in the collision transmit sequentially after the collision. To achieve this goal, the node that sends the CTS should include the addresses of the other nodes that have been involved in the collision. Note that the node that sends the CTS can decode the collided frames via the postamble.

Figure 2.6 shows Jain's fairness index for the system throughput under three protocols according to different transceiver ratios. As expected, the fairness of 802.11n is always better than that of the others, although it shows the lowest overall throughput performance due to performance anomaly. Differently, for 802.11n+ and 802.11mc, the MIMO nodes achieve higher throughput than SISO nodes, thus showing less fairness in terms of throughput. As verified in the result, the fairness of both protocols is heavily affected by the network configurations. First, in the (4,3,2,1) case where SISO nodes are the majority, a few MIMO nodes attain much higher throughput than the others; hence, the throughput disparity increases. For example, the four-antenna node can get a transmission opportunity whenever other nodes with low DoF acquire a medium. However, in the (1,2,3,4) case, the majority of the nodes in the network are MIMO nodes; hence, the concurrent transmission gains are distributed among the MIMO nodes, thus achieving better fairness. In addition, over all cases, 802.11mc shows better fairness compared with 802.11n+. This is because SISO and MIMO nodes can obtain throughput gain from the collision resolution.

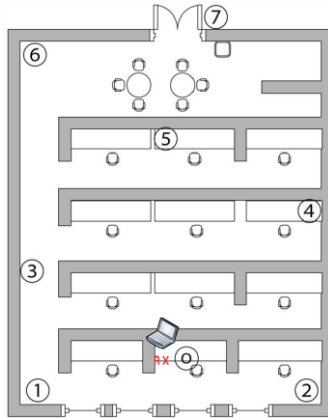


Figure 2.7: **802.11mc test environment.**

## 2.5 USRP Experiments

Using USRP and GNURadio [13], we conduct two types of experiments, i.e., micro benchmark and macro benchmark. We first evaluate the postamble decoding scheme, which is the main technique of 802.11mc by using the micro benchmark. Then, we emulate the system throughput of three protocols 802.11n, 802.11n+, and 802.11mc via the macro benchmark.

### 2.5.1 Micro Benchmark

#### 1) *Setup*

We implement five nodes on OFDM PHY using USRP/GNURadio. All nodes are equipped with USRP N210 on the SBX daughterboard and work on 10MHz bandwidth. Among five nodes, two nodes act as an 802.11mc MIMO receiver: one as a legacy SISO receiver and two as SISO transmitters. Two USRP nodes of the MIMO receiver are connected with a MIMO cable [13] and placed at location 0 in Figure 2.7 with the legacy receiver. For the transmitters, we randomly pick two locations.

To test for packet collision, we had to let both transmitting nodes start their transmission at the same time, but use different frames, i.e., a frame with a postamble and a non-postamble frame, so that their preambles are almost overlapped. Recall that 802.11 nodes generally are timely synchronized by beacon frames. To implement this for the USRP experiments, we simply make two transmitters act like one 2-antenna distributed MIMO transmitter and send the different frames. We use an OCTOCLOCK-G [13], which is a clock distribution system with an integrated GPS disciplined oscillator, to connect both transmitters and control their start times under distributed test environment. Some delays introduced by software radios are managed by USRP timestamp function similar to 802.11n+. As a result, in our experiment, the synchronization error is within one cyclic prefix.

Another possible concern of the USRP experiment is that two different signals sent from different locations may incur large power difference; hence, only the larger signal is captured by the receiver. Although the A/D converter of USRP N210 is 14 bits, the actual valid dynamic range is not as high as 84dB but is about 30dB. In our settings, we had to make online gain adjustments using the transmit power control to overcome the lack of automatic gain control. To do this, before starting the real experiment, we performed a test by sending signals concurrently to check whether all signals are properly host by the receiver. If not, we changed the power level of the transmitters until the receiver can host all signals in a stable manner.

Each frame has 500 bytes payload, and it is sent 10,000 times for each experiment. We implement the IAC decoding by MATLAB because it needs buffering. Hence, we first collect all traces and then perform the decoding procedure offline.

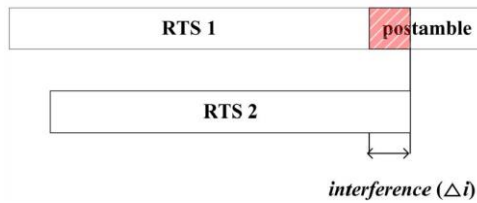


Figure 2.8: **Example of the misalignment.**

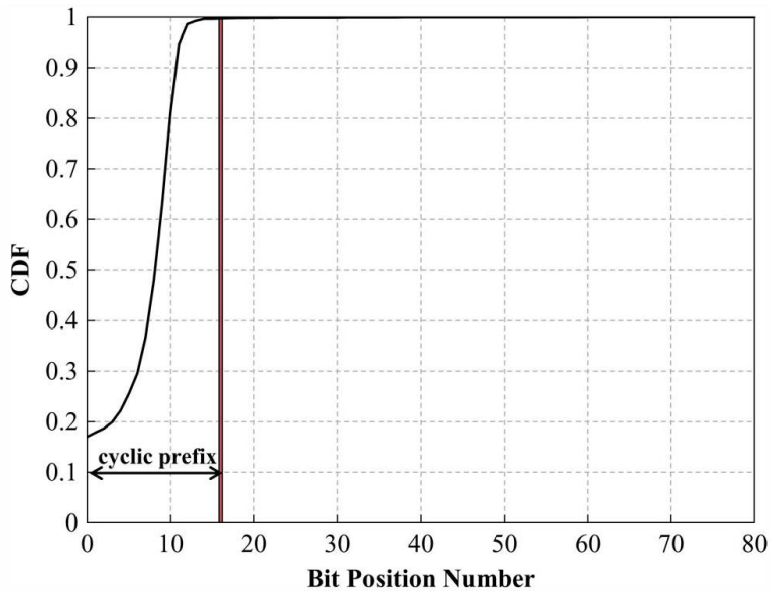


Figure 2.9: **Misalignment in two collided frames.** The portion of the postamble that is corrupted  $\Delta i$  is mostly covered by the cyclic prefix.

## 2) Number of Training Symbols

The length of postamble renders a tradeoff between the overhead and CSI estimation accuracy. Furthermore, if any collided frames overlap at the postamble portion, as shown in Figure 2.8, decoding may fail. In this experiment, we measure the length of the corrupted postamble caused by the misalignment  $\Delta i$  and show the result in Figure 2.9.

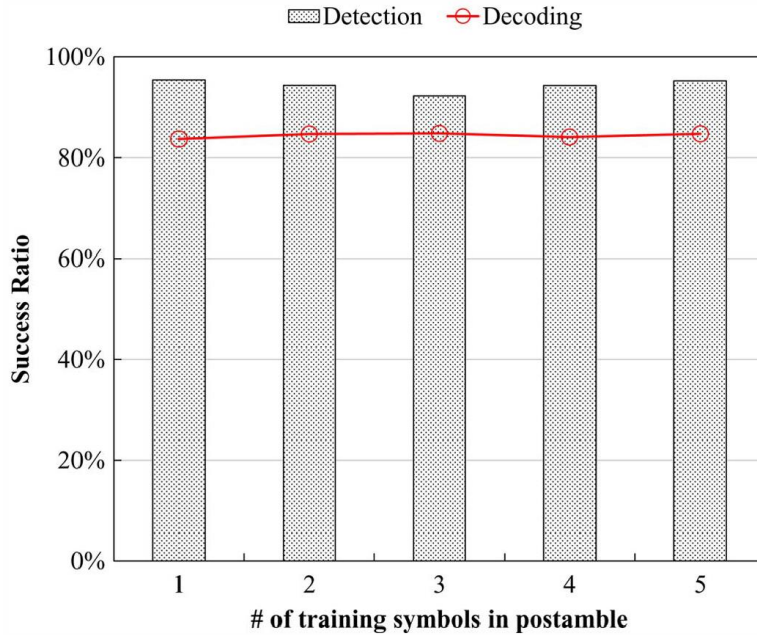


Figure 2.10: **Success ratio for varying the number of training symbols in the postamble.** This graph shows that, regardless of the length of training symbols, the collision detection and the decoding of collided frames achieve the success ratio of about 92% and 84%, respectively.

In our measurement, the portion of the postamble that is corrupted  $\Delta i$  is mostly covered by the cyclic prefix. From this result, we observe that the misalignment caused by the collision does not affect the postamble. Recall that we consider the collisions invoked by more than two nodes with the same backoff counters. Next, we vary the number of training symbols in the postamble from 1 to 5 and conduct the experiment under the same environment as the first experiment. We measure two success ratios, i.e., one to detect a collision and the other to decode collided frames. From the result in Figure 2.10, we see that the proposed scheme achieves the success ratio of 92% and 84% on average for the detection and for the decoding performance, respectively.



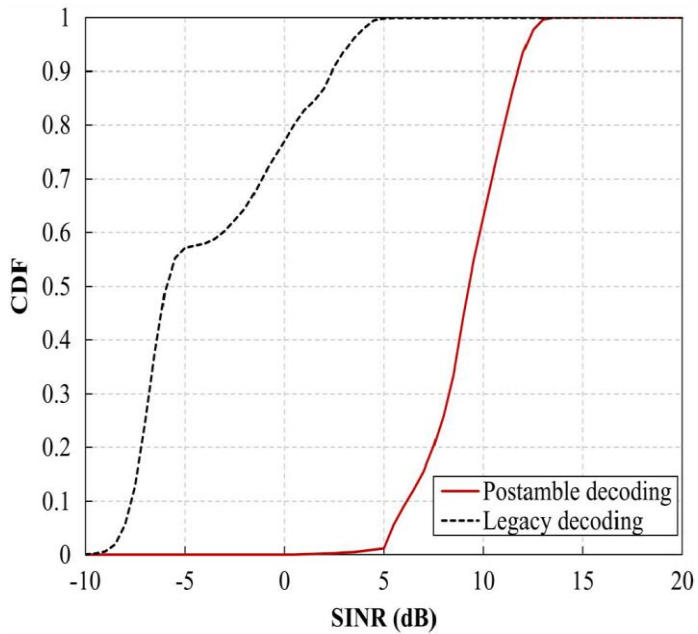


Figure 2.11: **SINR enhancement of postamble decoding.** Postamble decoding achieves the SINR gain of 15.5dB on average.

### 3) SINR Performance

The successful postamble-based decoding significantly improves the frame SINR. We compare the SINR gain obtained from the postamble-based decoding with the legacy decoding. We use the same setting as the first experiment unless stated otherwise. We measure and plot each SINR of the two decoding cases, the postamble decoding and the legacy decoding, as shown in Figure 2.11. As expected, the SINR of the legacy decoding is always below 7dB and a median of  $-6$ dB, whereas postamble decoding maintains an SINR of up to 15.5dB and a median of 9.5dB.

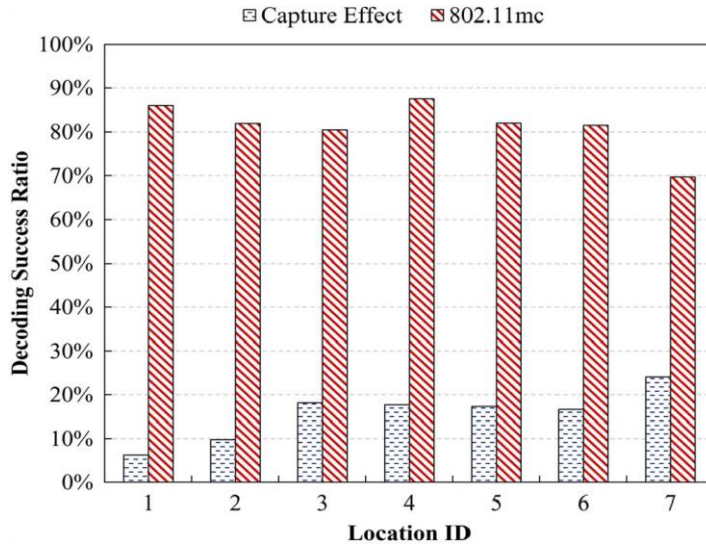


Figure 2.12: **Decoding success ratio of two schemes.** 802.11mc decodes collided frames with higher probability than the capture effect. We note that the performance of the capture effect can be varied according to network environment factors, such as received signal power and location of nodes.

4) *802.11mc vs. Capture Effect:* Although 802.11mc resolves the collision problem, the physical layer capture effect gives a similar result [37]; it enables a receiver to decode one of collided frames if the received signal strength of one frame is significantly higher than the other. Therefore, we study the effect of capture effect in this set of experiments. To evaluate the case where capture effect occurs, we put one sending node at location 1 and change the other node’s position from location 1 to location 7 in Figure 2.7.

Figure 2.12 shows the decoding success ratio of each mechanism. In this result, the capture effect achieves decoding success ratio of 24% when one of sending nodes locates outside the door because the performance of the capture effect is related to the difference of received power between two frames. On the other hand, in 802.11mc, all of the decoding success ratios are above 80%, except for the case of location 7. From this result, we demonstrate that the postamble-based decoding guarantees a higher decoding success ratio than the capture effect. We note that the capture effect cannot artificially control the frame to be protected under the collision, as opposed to 802.11mc.

## 2.5.2 Macro Benchmark

### 1) *Setup*

Next, we run some experiments to compare 802.11mc with 802.11n and 802.11n+. Unfortunately, conducting a real-time experiment by implementing the MAC functionalities (e.g., CSMA, RTS/CTS, and ACK) in USRP/GNURadio is not simple; when software radios exchange signal samples between the host and the RF front end, it incurs high latency [38]. Moreover, when the MIMO receivers recover the collided frame via the postamble, it needs to buffer some previously received samples for the recovery, and this would be even more difficult to implement in real time. Recall that we performed the postamble decoding offline in the previous set of experiments due to these limitations. Therefore, we construct a macro benchmark inspired by [16] and [20] to perform the throughput comparison in a real setup.

The experiment consists of three steps, i.e., interference map generation, USRP transmission, and emulation. In the first step, we make an interference map to capture the network contention of the real world. In the same settings shown in Figure 2.7, we place laptops each with a MadWiFi driver [39] in locations 1–6, and let each laptop in locations 1, 2, and 5 transmit packets to laptops in location 3, 4, and 6, respectively. Since the laptops have no MIMO capability, for transmitters, we simply let them send different size packets (500-, 750-, and 1500-byte payload for each type). We log all trace information (e.g., time, delivery ratio, and SNR) and repeat this work again for the case of RTS/CTS.

After obtaining the interference map, we place the USRP nodes in each location again; the nodes in locations 1 and 3 are SISO; the nodes in locations 2 and 4 are two-antenna MIMO; the nodes in locations 5 and 6 are three-antenna MIMO. Then, we mimic the same behavior with the interference map. We let the USRP nodes transmit alone frames, i.e., frames without collisions, as many as the number of non-collided frames of the interference map and then let them transmit together for the packet loss trace. Here, we assume that the packet loss occurs due to the collision only. Since the trace includes timing information, we can infer which nodes are involved in a certain collision. All data frames are transmitted at a certain data rate using modulations BPSK, QPSK, 16-QAM, and 64-QAM based on the SNR traces. Note that the USRP nodes in our testbed work only at 10MHz bandwidth; hence, the achievable data rates

are smaller than those of real world. To implement the concurrent transmission in 802.11n+ and 802.11mc, we use the code provided in [5]. The lightweight RTS/CTS scheme [24] was not applied.

Next, we emulate each protocol using the traces. We carefully account for the time-related intervals (e.g., SIFS, DIFS, and random backoff functionality) and ACK packets in our emulation. In addition, since there are a small number of competing nodes, we vary the CWMin from 3 to 15 during the experiment. Otherwise, this will result in an unrealistic situation where frame collisions rarely occur.

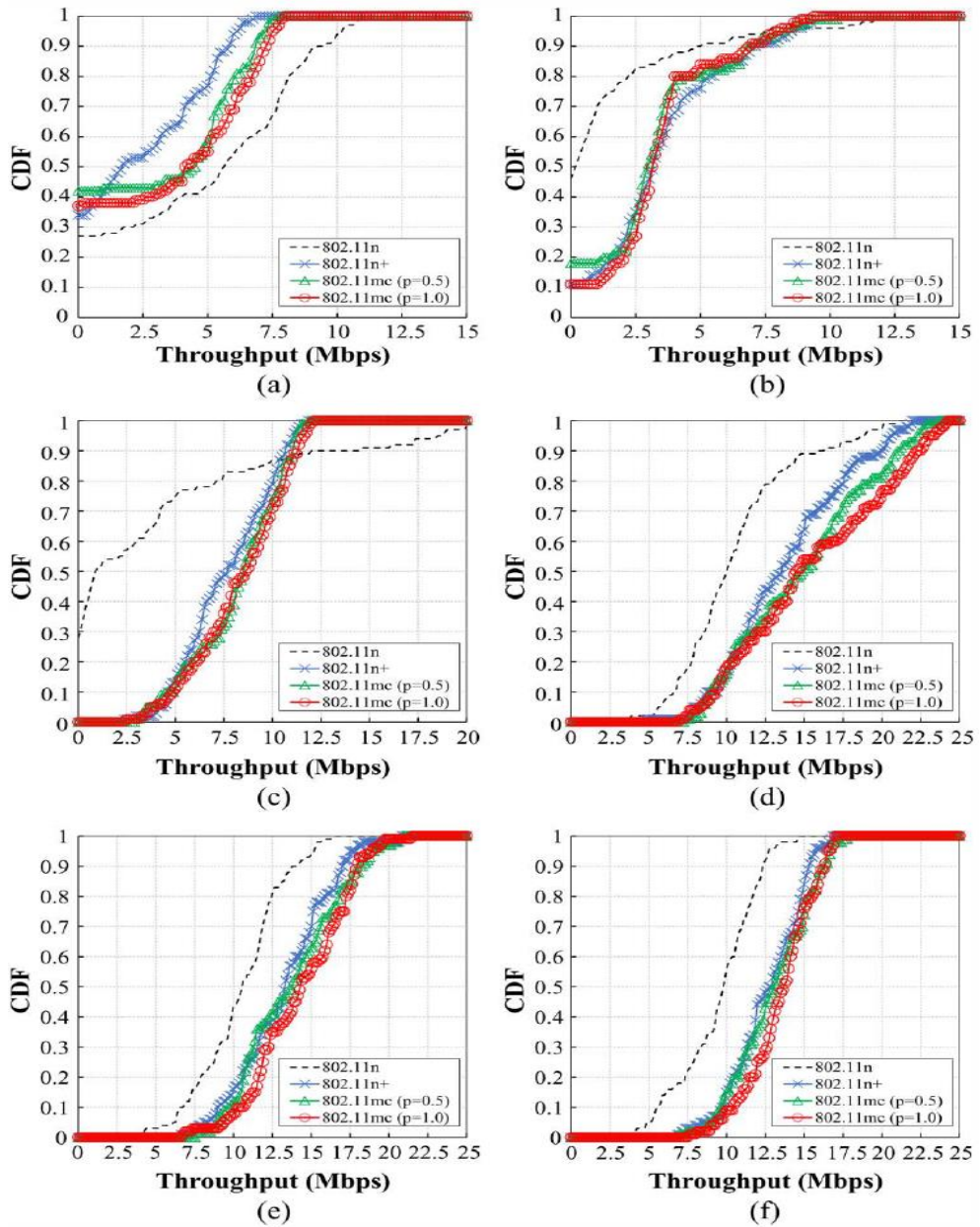


Figure 2.13: **Throughput comparison in a macro benchmark.** When CWMin is 3, the gains of 802.11mc are about 47% and 12% on average, compared with 802.11n and 802.11n+, respectively. (a) SISO pair (CWMin = 3). (b) 2 × 2 MIMO pair (CWMin = 3). (c) 3 × 3 MIMO pair (CWMin = 3). (d) System (CWMin = 3). (e) System (CWMin = 7). (f) System (CWMin = 15).

## 2) Throughput Comparison

Figure 2.13 plots the throughput CDF of each system: 802.11n, 802.11n+, and 802.11mc ( $p = 0.5$  and  $p = 1.0$ ). In terms of system throughput, the gain of 802.11mc increases as CWMin decreases since the collision rate will increase. As a result, when CWMin is 3, the throughput gains of 802.11mc are about 47% and 12% on average, as compared with 802.11n and 802.11n+, respectively.

These gains come mainly from the high channel utilization due to the collision resolution, which gives all the nodes more transmission opportunities. In particular, for MIMO nodes, the gain of the concurrent transmission increases significantly. For example, if the channel time of SISO increases by  $\Delta t$ , then the four-antenna MIMO node may obtain up to  $3r\Delta t$  throughput, where  $r$  is the data rate of SISO.

Since 802.11n+ and 802.11mc both use the RTS/CTS exchange, the performance could be limited by its overhead. In particular, it decreases the throughput of the SISO pair, as opposed to the 802.11n without RTS/CTS, as shown in Figure 2.13 (a). In our result, without the lightweight handshake, the SISO pair's throughput in 802.11n+ is reduced by 4Mbps, whereas that in 802.11mc is reduced by only 1Mbps on average, as compared with 802.11n. Note that we can adopt overhead reduction techniques like the lightweight handshake. In addition, we see that the optimal postamble probability enhances the performance of 802.11mc in all cases, as shown in Figure 2.13.

Table 2.3: NS-2 Simulation Parameters

Parameter	Value
CWMin	15
Payload	1500 Bytes
$q_2$	0.5
N	20
P	1
Data modulation	QPSK

## 2.6 NS-2 Simulations

### 2.6.1 Setting

The experiments based on USRP/GNURadio cannot give the detailed and large-scale evaluations to test the complexity and dynamics of node contentions. Therefore, we implement the three MAC protocols, i.e., 802.11n, 802.11n+, and 802.11mc, on the NS-2 simulator to analyze and compare the performance.

We set a random topology with  $2N$  nodes and configure so that they can sense each other. The nodes have IDs from 1 to  $2N$ , and we make two nodes as one transceiver pair. In this setting, the node  $i$  with the odd number ID sends data to the node  $i + 1$ . To construct a heterogeneous MIMO network, we set half of transmitters as MIMO nodes and the others as SISO nodes. We assume that all MIMO nodes have just two antennas (i.e.,  $M = 2$ ). We set the simulation parameters to the default values in Table 2.3 and the 802.11n standard [1]. Each transmitter generates constant bit rate traffic every 1ms, and the simulation time is set to 20s. We run each simulation 100 times and measure the throughput and throughput gain of 802.11mc over other protocols 802.11n and 802.11n+.

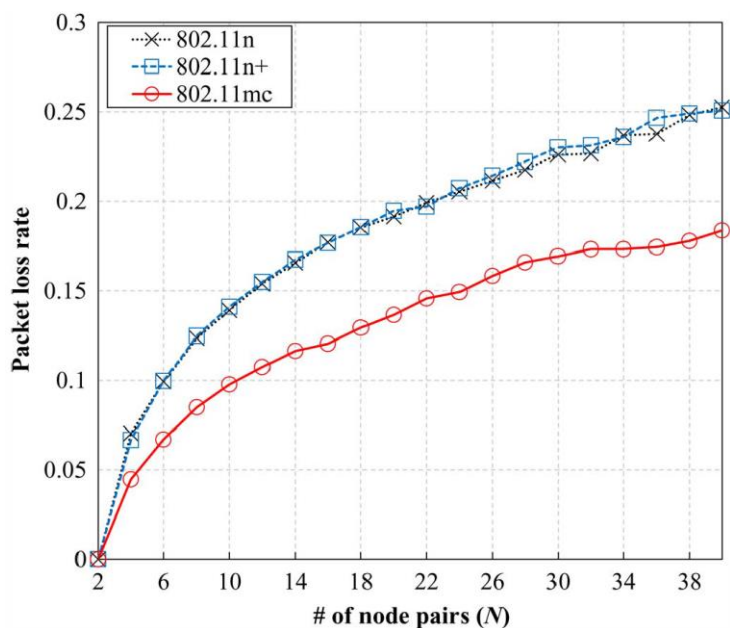


Figure 2.14: **Packet loss rate due to collision.** 802.11mc lowers the packet loss rate by up to 7% compared with the other protocols.

## 2.6.2 Packet Loss Rate due to Collision

In Figure 2.14, we measure the packet loss rate due to collision as the number of nodes increases. Recall that 802.11mc resolves the collision to avoid packet losses. We examine that 802.11mc constantly lowers the packet loss rate by up to 7%, as compared with the other protocols. Since 802.11n and 802.11n+ cannot resolve the collision, they almost show identical performance. At first glance, the packet loss rate difference between 802.11mc and 802.11n+ may seem small, but we can see that 802.11mc significantly outperforms both 802.11n and 802.11n+, as will be shown in the later results. Recall that 802.11mc nodes obtain the gain from two sources, namely, time domain and spatial domain. Clearly, the 802.11mc will achieve better channel utilization due to the decreased loss rate. This gain of increased opportunities is further amplified by the concurrent transmissions by MIMO nodes. As a result, the relatively smaller decrease in loss rates translates into a larger increase in throughput.



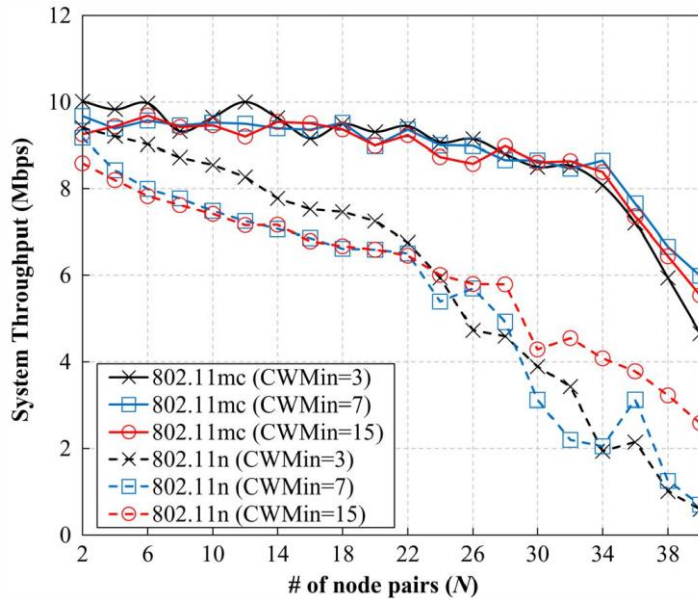


Figure 2.15: **Impact of CWMin.** 802.11mc is not sensitive to the CWMin compared with legacy 802.11n. This result indicates that the performance of 802.11mc is influenced by the number of contending nodes more than the CWMin.

### 2.6.3 CWMin

In this set of simulations, we vary the CWMin as 3, 7, and 15 to see how 802.11mc reacts for various collision probabilities caused by the CWMin. Note that in general 802.11 networks, a smaller CWMin will give less channel idle time but will increase the collision probability.

As shown in Figure 2.15, 802.11n is significantly influenced by the CWMin as the number of nodes increases. When the CWMin is 3, the throughput of 802.11n results in the best performance until the number of pairs reaches 20. Then, it decreases due to the increased number of collisions. Meanwhile, the performance of 802.11mc is steady with variable CWMin. This is because the collision resolution of 802.11mc makes the effect of CWMin relatively weak. This result indicates that the performance of 802.11mc is determined by the number of contending nodes more than the CWMin. In addition, the gap becomes even larger as the number of node pairs increases, which shows the robustness of 802.11mc against the frame collision problem.

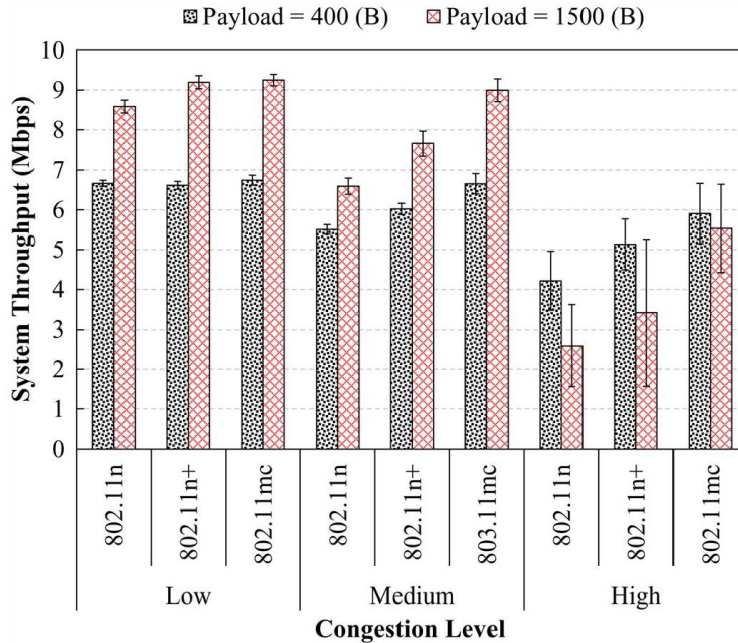


Figure 2.16: **Throughput versus data size with various congestion levels.** The throughput of each protocol is measured for various congestion levels. We use two types of payloads, namely, small (400 bytes) and large (1500 bytes).

## 2.6.4 Data Size

Similar to 802.11n+, the RTS/CTS handshake of 802.11mc may cause overhead, particularly when the data size is very small. Furthermore, 802.11n does not employ RTS/CTS since this is the general case and it is only used as an option. We vary the data size by 400 and 1500 bytes. Additionally, we define three network congestion levels, i.e., “Low,” “Medium,” and “High” when the number of pairs is 2, 20, and 40, respectively.

First, as shown in Figure 2.16, for all protocols, the performances of large data are generally better than those of the small data size, except the case of the highly congested network. In particular, when the number of pairs is 40, i.e., the congestion level is “High,” the throughput of the small data increases. This is because when a collision occurs, the large data obviously spend more channel time, plus they have to defer their retransmission much longer than that of the small data. Second, both

802.11n+ and 802.11mc overcome the overhead of RTS/CTS by the MIMO gain and outperform legacy 802.11n. In particular, 802.11mc always obtains a higher throughput than 802.11n+ due to its collision resolution. As a result, the gain of 802.11mc over 802.11n+ becomes much larger as the collision probability increases; it increases from 1% to 62% as the congestion level becomes high.

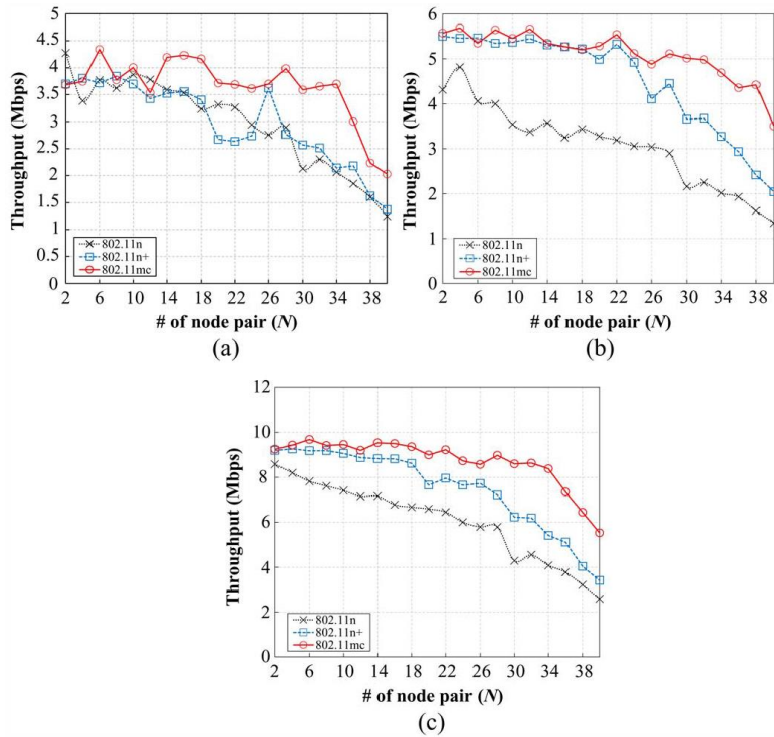


Figure 2.17: **Throughput according to the number of pairs.** As the collision probability increases, 802.11mc achieves higher throughput due to additional transmission opportunities of the transmitting nodes. (a) SISO node. (b) MIMO node. (c) System.

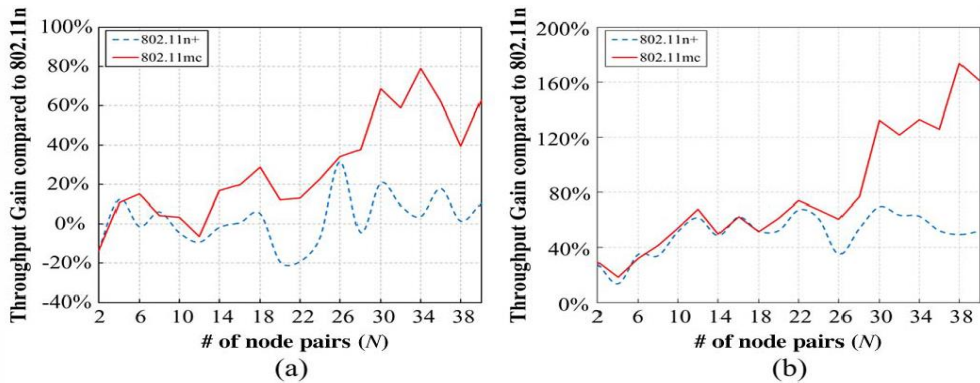


Figure 2.18: **Throughput gain of 802.11mc and 802.11n+ over 802.11n.** Different from 802.11n+, the throughput of SISO nodes also increases in 802.11mc. As a result, the throughput gain of 802.11mc increases by up to 79 % and 174 % even in a heavily congested network. (a) SISO node. (b) MIMO node.

### 2.6.5 Number of Node Pairs ( $N$ )

We compare the throughput of the three protocols according to the number of pairs in Figure 2.17. Each graph shows the SISO, MIMO, and system throughput, respectively.

As shown in Figure 2.17, the throughput of the three protocols decreases as the number of pairs increases. However, the throughput of both 802.11mc and 802.11n+ fall smoothly because of the small size of the RTS/CTS. In particular, 802.11mc shows better throughput than 802.11n+. The main reason is that while only MIMO transmitters improve the system performance of 802.11n+, 802.11mc gives all the nodes additional transmission opportunities so that the system achieves higher channel utilization.

In order to clarify the result, we provide the throughput gain of 802.11n+ and 802.11mc over 802.11n in Figure 2.18. The result of 802.11n+ shows that the gain of SISO nodes does not increase but vibrates around 0%, whereas that of MIMO nodes increases up to 60%. This means that there are no benefits for SISO nodes in 802.11n+. Meanwhile, in 802.11mc, both gains of nodes, regardless of their types, become high with the collision probability. As a result, the throughput gain of 802.11mc over 802.11n for each type of node increases by up to 79% and 174%, even in a heavily congested network.

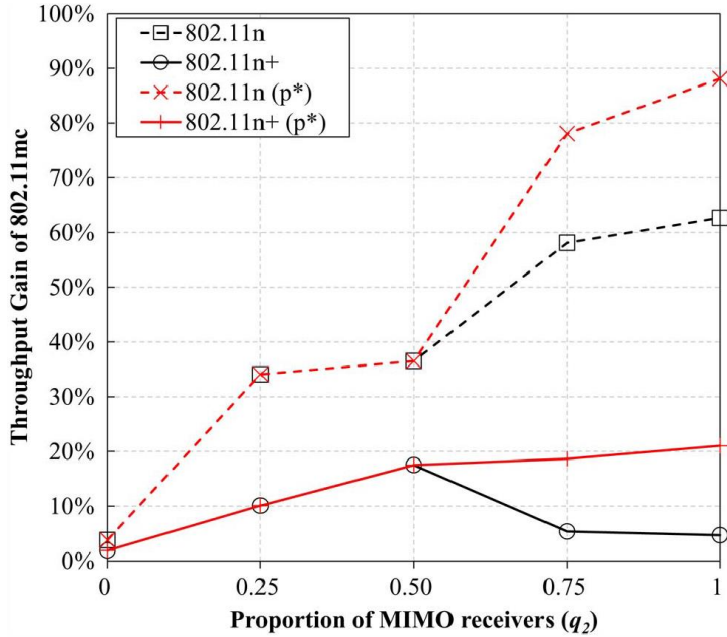


Figure 2.19: **Throughput gain of 802.11mc over other two protocols according to the proportion of MIMO receivers ( $q_2$ ).** 802.11mc provides higher throughput gain as the proportion of MIMO receivers increases. We note that the system with non-optimal  $p$  could create more unsolvable collision cases than that with optimal  $p$ . Nevertheless, the throughput of 802.11mc is higher than that of 802.11n+ in all cases.

## 2.6.6 Proportion of MIMO Receivers ( $q_2$ )

The proportion of MIMO receivers ( $q_2$ ) affects the system performance of 802.11mc because only MIMO receivers can resolve the collided frames. In order to evaluate the performance to vary the proportion of MIMO receivers, we conduct a simulation varying  $q_2$  from 0 to 1. In this simulation, we set  $p$  to 1, except for the optimal case.

From the result in Figure 2.19, we observe that both gains of 802.11mc over 802.11n and 802.11n+ increase with  $q_2$  (maximum 88% and 21%). In particular, when  $p = 1$ , the gain over 802.11n+ decreases after the proportion reaches 50% because the non-optimal  $p$  could create some unsolvable collision cases. Note that the optimal value of  $p$  should proportionally decrease with the increase in MIMO receivers, as shown in Figure 2.5. Nevertheless, the throughput of 802.11mc is consistently higher than that of 802.11n+ in all cases.

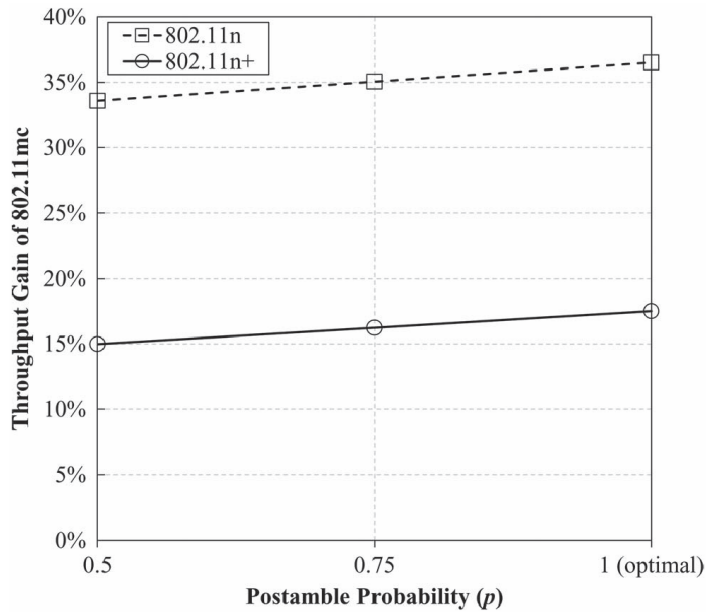


Figure 2.20: **Throughput gain of 802.11mc over other two protocols according to the postamble probability ( $p$ ).** With the optimal postamble probability, 802.11mc improves its throughput gain by 2%.

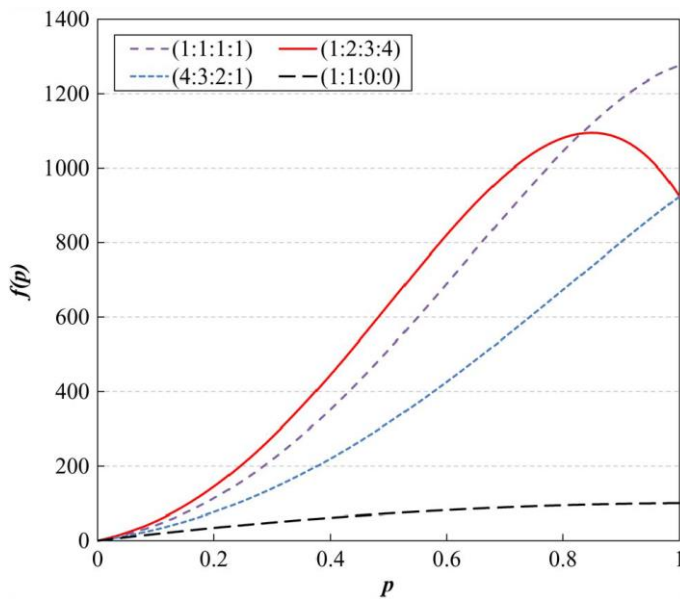


Figure 2.21:  **$f(p)$  under different network configurations.** The effect of  $p$  depends on the network configuration.

### 2.6.7 Postamble Probability ( $p$ )

The postamble probability  $p$  is very important to improve the system performance in 802.11mc because the number of resolvable collision cases depends on it. In order to investigate the performance for the postamble probability, we measure the throughput gain of 802.11mc to vary  $p$  from 0.5 to 1. We set the number of transmitters, the proportion of MIMO receivers, and optimal  $p$  to 20, 0.5, and 1, respectively.

As shown in Figure 2.20, the system gain of 802.11mc increases, as  $p$  is close to the optimum. As a result, 802.11mc obtains the throughput gain of up to 37% and 17% over 802.11n and 802.11n+, respectively.

The effect of  $p$  depends on the network configuration, as shown in Figure 2.21. First, as the number (or ratio) of MIMO nodes increases, 802.11mc generally increases the number of resolvable collisions with larger  $p$ . However, as the number of MIMO nodes further increases, the optimal  $p$  becomes less than 1, e.g., the (1:2:3:4) case. Second, if the antenna numbers are less diverse, e.g., the (1:1:0:0) case, the effect of  $p$  on the resolvable collisions becomes much less evident. This is why  $p$  shows less impact in Figure 2.20. Note again that in any case, the optimal  $p$  ranges from 0.5 to 1. Even if we adopt a non-optimal  $p$ , 802.11mc still provides a minimum level of performance gain over other protocols due to the fact that it can still resolve a certain number of collisions, albeit limited, and enable concurrent transmission.



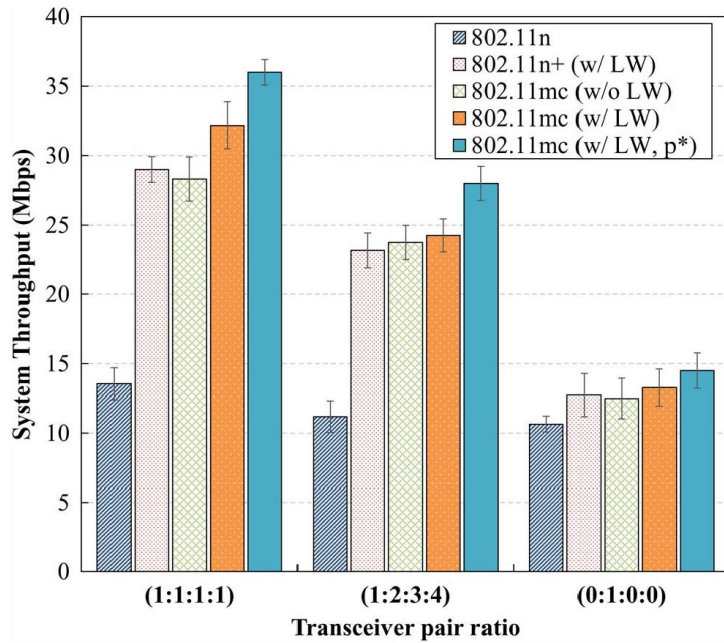


Figure 2.22: **System throughput comparison under different network configurations.** The  $p$  and optimal  $p$  are set to 0.5 and 1, respectively, for (1:1:1:1) and (1:2:3:4) and set to 1 and 0.5, respectively, for (0:1:0:0). “LW” denotes the lightweight RTS/CTS. The result of 802.11n+ without the lightweight handshake is omitted. Additionally, we use receiver-based auto rate [33] as a rate adaptation algorithm.

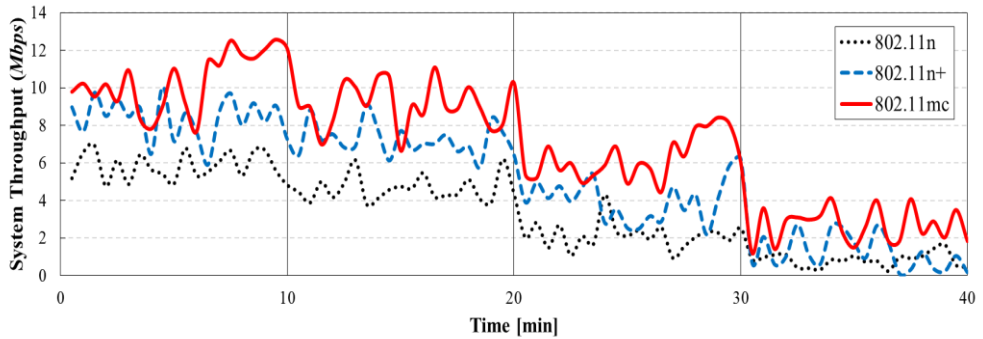


Figure 2.23: **System throughput comparison under dynamic network configurations.** 10 transmitters join the network every 10 minutes. The  $p$  is set to 0.5.

## 2.6.8 Performance in Dynamic Network Configurations

We evaluate the throughput of each protocol under three different network configurations and illustrate the result in Figure 2.22. Except for the case of (0:1:0:0), the simulation runs with four types of nodes, from SISO to four-antenna MIMO nodes. We also evaluate the performance of the protocols with and without the lightweight handshake scheme.

First, when a network lacks the diversity in terms of antennas, e.g., (0:1:0:0), the throughput gain of 802.11mc decreases. In such a homogeneous MIMO network, 802.11n+ and 802.11mc do not need to enable multiuser transmission since each node can exploit the full DoFs. Thus, the only source of gain over 802.11n becomes the increased channel utilization for 802.11n+ and 802.11mc, respectively. This makes both protocols obtain a smaller gain over 802.11n in a homogeneous network than in a heterogeneous network. Nevertheless, 802.11mc still shows better performance than the others due to the collision resolution, as shown in Figure 2.22. In addition, we believe that the future wireless network will become more populated by heterogeneous devices in terms of the number of antennas.

Second, the network configuration significantly affects the system performance; for 802.11n+ and 802.11mc, the performances of the (1:1:1:1) case are better than those of the (1:2:3:4) case. This is because the reduced number of SISO nodes reduces the gain from multiuser transmission in the case of (1:2:3:4); the relatively long transmission time of SISO (low-DoF) nodes gives more time to MIMO (high-DoF) nodes for simultaneous transmission. Third, as previously discussed in Section 2.4.5, the lightweight handshake can improve the performance of 802.11mc and 802.11n+. In the case of the ratio of (1:1:1:1), the system throughput is increased by about 14% from the lightweight handshake. Additionally, optimal  $p$  also enhances the throughput of 802.11mc. Compared with the result in Figure 2.20, we can see the impact of optimal  $p$  more clearly, which means that the postamble probability affects the network more effectively in the case of high diversity in the number of antennas.

Figure 2.23 shows the system throughput comparison under dynamic network configurations. In this setting, we let 10 transmitters (5 for SISO, 5 for MIMO) join the network every 10 minutes. As shown in the figure, we can clearly see that 802.11mc outperforms other protocols even in random network environment.

## 2.7 Conclusion

In this chapter, we have proposed a MAC protocol, i.e., 802.11mc, which addresses the frame collision problem. The 802.11mc protocol uses the postamble to realize the IAC for decoding collided frames so that all nodes, regardless of the number of their antennas, obtain additional transmission opportunities, and the system achieves better channel utilization. We give an analytic model to obtain the optimal mc-RTS append probability  $p$  that achieves the maximum performance. To evaluate the performance of 802.11mc, we conduct USRP/GNURadio-based experiments and the extensive NS-2 simulations. The results show that 802.11mc achieves a throughput gain of up to 88% and 21% over 802.11n and 802.11n+, respectively.

# CHAPTER III

## User Selection for MU-MIMO Transmission

### 3.1 Introduction

In this chapter, we present an MU-MIMO MAC protocol for Wi-Fi systems, called 802.11ac+, which provides a novel and practical user scheduling solution. Our idea is very intuitive: perform the user scheduling during the CSI feedback phase. To do this, the AP broadcasts channel information about previously scheduled users by appending it to a poll frame. We refer to this channel information as a *channel hint*. Then, users calculate their Effective Channel Gains (ECGs) from the channel hint, and the user with the largest gain actively sends a CSI report back to the AP. Upon receiving the CSI report, the AP includes the user in the multi-user transmission schedule and repeats this process while its DoF constraint is satisfied. As a result, the performance of 802.11ac+ is comparable to other user scheduling heuristics, with the key difference that it requires a much smaller amount of CSI feedback.

The design of 802.11ac+ is challenging for the following reasons. First, the channel hint should be well designed to balance between overhead and efficiency. In order to reduce the overhead, the channel hint should be succinct, but at the same time, it should include all channel information of already scheduled users such that the remaining users can estimate their ECGs. To deal with such a trade-off, we propose an efficient channel hint broadcasting mechanism. More specifically, an AP extracts the effective channel vector of the last scheduled user from the received CSI reports, and uses it as a channel hint. The second challenging issue comes from the fact that users hardly know how much better their ECGs are in a fully distributed manner, which leads us to adopt a contention for users to get a feedback opportunity. Due to the nature of contention, there may be a frame collision if more than one user sends a CSI report at the same time. To address the collision problem, we use a delayed feedback approach, where users delay their CSI feedback according to their ECGs: the bigger the ECG a user has, the faster the CSI report will be. To improve the performance, we focus on maximizing the probability of feedback success. Additionally, we extend our user selection algorithm to well-known fair scheduling protocols, Round-Robin (RR) and Proportional-Fair (PF), to achieve fairness among users.

To evaluate the performance of our approach, we implement the 802.11ac+ and its fair scheduling protocols on a MATLAB simulator. The extensive trace-driven simulation results show that 802.11ac+ obtains much higher throughput gain than 802.11ac and a MAC protocol employing a well-known user scheduling algorithm. Also, we demonstrate that two fair scheduling protocols of 802.11ac+ give a much better throughput fairness than 802.11ac, especially when users experience different channel qualities.

The remainder of this chapter is organized as follows. Section 3.2 provides the related work and we next introduce the background of this chapter. We describe the 802.11ac+ mechanism in greater detail in Section 3.4. In Section 3.5 we discuss the fair scheduling protocols of 802.11ac+, and Section 3.6 shows the performance evaluation. We finally conclude this chapter in Section 3.7.

## 3.2 Related Work

We survey the research results on the user scheduling schemes and channel feedback overhead reduction techniques related to 802.11ac+.

### User scheduling schemes

User selection has been highlighted since it can improve the MU-MIMO performance significantly. In user scheduling, it is challenging and often impractical to determine the optimal user set due to the large search space. Therefore, some protocols based on heuristics have been proposed. Many publications study the problem of maximizing sum capacity [7], [10], [11], [40]. ZFS [7] chooses a user that maximizes the sum-capacity. In GWC-ZFBF [8], the AP chooses a user with the largest channel power gain. SUS [9] selects one user in each round by exploiting channel orthogonality between users. However, the selected users cannot guarantee that they always increase the sum rate. GUSS [11] considers 'delete' and 'swap' operations to guarantee a positive increment of channel capacity in each selecting round. Jin *et al.* propose a volume metric as the product of diagonal elements of an upper-triangular matrix by performing QR factorization to the selected user channels [10]. Shen *et al.* [40] propose and compare two algorithms for both approaches with block diagonalization [41] which is a generalized concept of channel inversion.

OPUS [12] is a user selection scheme for MU-MIMO Wi-Fi systems that bears the most similarity to our work. In OPUS, users estimate their potentials (e.g., SINR) in each round to boost the capacity and initiate a distributed feedback contention. The potential measurement scheme closely follows the main idea of the SUS (i.e., the largest projection power). However, the result based on the projected norm may not satisfy the maximum sum-capacity. Also, the time domain contention employed in OPUS may cause non-negligibly high overhead.

In addition to the sum rate maximization, the rate balancing problem aimed at maximizing the throughput subject to the constraint that the rates of the different users need to have certain fixed ratios has also been considered in many literatures [42], [43]. Hellings *et al.* propose a gradient projection-based solution [42], and Guthy *et al.* propose a user classification scheme based on perturbation analysis [43], in order

to handle the rate balancing problem. Along this line, Lima *et al.* consider a frequency allocation problem for MIMO-OFDMA networks [44]. Unfortunately, these schemes come with a non-negligible performance loss due to the excessive feedback overhead.

### **Channel feedback overhead reduction techniques**

One way of overcoming the CSI overhead is to use the compression techniques for reducing CSI feedback bits. Codebook and quantization are already adopted in LTE and MIMO-based Wi-Fi systems [1-3], [45], [46]; however, selecting the optimal quantization level is still an open problem. In Wi-Fi systems, it is proposed to compress the CSI report along three dimensions: time, frequency and quantization level. Even though compression is used, it is reported that the feedback can take about 25 times longer than the data transmission time when sending a small packet with a high data rate [6]. A vast literature of work has aimed to design an efficient CSI quantization mechanism [47-51]. Recently, compressive sensing has also been used for feedback reduction in MIMO communications [52], [53]. The downside of compressions is, as expected, a throughput loss of MU-MIMO transmission: fewer bits may offer diminishing returns.

In addition to the compression, the reduction of feedback loads by adaptively sending feedback can be used. One possible solution is to allow only users whose signal quality (e.g., SNR, Carrier-to-Noise Ratio (CNR)) is higher than a pre-defined threshold to report their CSI [6], [54-57]. However, these approaches may fail to obtain higher effective channel gain because their CSI reports are sent without considering the relationship between user channels.

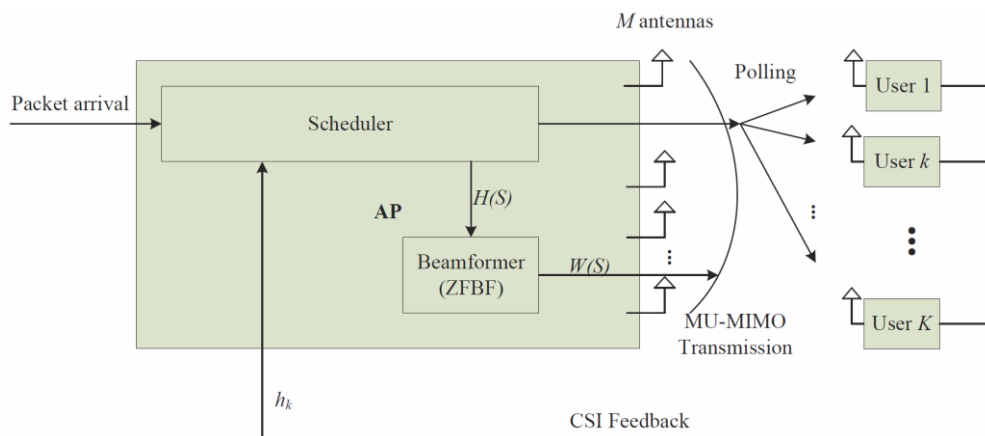


Figure 3.1: MU-MIMO downlink system with the  $M$ -antenna AP and  $K$  single-antenna user stations.

## 3.3 Background

### 3.3.1 System Model

In this chapter, we consider a single Basic Service Set (BSS) Wi-Fi network, where an  $M$ -antenna AP and  $K$  single-antenna user stations communicate with each other, as shown in Figure 3.1. We assume that MIMO channels satisfy the i.i.d. Rayleigh fading condition: the real and imaginary components of  $\mathbf{h}_{\mathbf{k}m}$  are i.i.d. Gaussian with unitary variance and zero mean, where  $\mathbf{h}_{\mathbf{k}m}$  is a channel gain from the  $m$ th antenna of the AP to the  $k$ th user. Then, we can characterize the channel of user  $k$  from the AP, i.e.,  $\mathbf{h}_{\mathbf{k}}$ , as a zero mean complex Gaussian channel vector.

In MU-MIMO, the multi-antenna transmitter, e.g., AP, uses precoding to send multiple data streams between several users at a given instant. Similar to most recent schemes, we incorporate Zero-Forcing Beamforming (ZFBF) as the precoding strategy, since it effectively removes the mutual interference among concurrent transmissions by using a low-complexity precoding matrix computation. In ZFBF, the precoding vector of one user is selected to be orthogonal to the channel vector of



the other user. In particular, the precoding matrix for the transmission group  $S$ , denoted by  $W(S)$ , is obtained as:

$$W(S) = H(S)^\dagger = H(S)^*(H(S)H(S)^*)^{-1} \quad (11)$$

where  $(\cdot)^\dagger$ ,  $H(S)$  and  $H^*$  stand for a pseudo-inverse, the channel matrix of  $S$ , and the conjugate transpose of  $H$ , respectively.

Let  $X(S)$  be the signal vector to be transmitted for  $S$ . Then, the precoded signal vector, denoted by  $X(S)' = W(S)X(S)$ , has an average power constraint of  $P$  and  $\mathcal{E}[\cdot]$  are the maximum transmitting power of the AP and the expectation operator, respectively. The sum-capacity for  $S$ , denoted by  $C(S)$ , is modeled as:

$$C(S) = \max_{P_j: \sum_{j \in S} \gamma_j^{-1} P_j \leq P} \sum_{j \in S} \log(1 + P_j) \quad (12)$$

, where  $\gamma_j = \frac{1}{\|w_j\|^2}$  is the effective channel gain of  $j$ th user in  $S$  and  $w_j$  is an element of  $W(S)$ .

### 3.3.2 User Selection

The main objective of the user selection problem is to maximize the sum-capacity for a user group  $S$ :

$$\underset{S \subseteq \{1, \dots, K\}: |S| \leq M}{\text{maximize}} C(S). \quad (13)$$

Generally, (13) is accomplished by taking two steps: optimal user selection and power allocation. Note that the water filling is well known for the optimal power allocation approach [26]. In this chapter, we assume the equal power distribution scheme for simplicity.

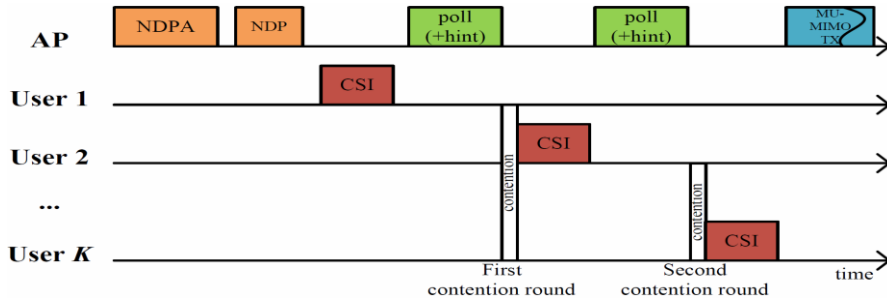


Figure 3.2: **Operation example of 802.11ac+**. Both AP and users participate in scheduling decisions. Except for the first scheduled user, all users should send a feedback report through a contention. If the AP fails in reception of any CSI reports, then it does not go to the next contention round, and finishes the feedback procedure. Therefore, there could be maximum  $(M - 1)$  contention rounds in the case of  $M$  AP antennas.

## 3.4 802.11ac+

### 3.4.1 Overview

As explained earlier, we do not separate CSI feedback from the user scheduling procedure. This is the key difference between the user scheduling in 802.11ac+ and other approaches. Figure 3.2 illustrates the main idea. At the beginning, an AP polls the first user (i.e., User 1 in this example) and it replies with its CSI. Then, the AP immediately joins User 1 to the scheduled user set  $S$ . We note that the first user is selected by the AP's queuing policy. By using the first CSI, the AP makes a channel hint, and it broadcasts a polling frame with the channel hint to all users. Upon receiving the poll, users compute their ECGs from the channel hint. Then, the user with the highest gain sends its CSI report to the AP through a contention. In this example, User 2 has the largest channel gain so it can feed back CSI for the second time. This step repeats until the AP successfully receives  $M$  CSI reports or the feedback timeout is triggered.

The intuition behind our user scheduling method is that users actively participate in scheduling decisions, unlike in most user scheduling algorithms, which are executed at only the multi-antenna transmitters. As a result, user scheduling in 802.11ac+ can limit the number of sounding exchanges to the number of AP antennas, while still

keeping the scheduling gain. In the next section, we introduce two main mechanisms to realize 802.11ac+: channel hint broadcasting and active CSI feedback.

### 3.4.2 Channel Hint Broadcasting

#### 1) *Effective Channel Gain*

Since the user scheduling in 802.11ac+ is based on harnessing the ECG, we first look into the concept of it. For a user channel  $\mathbf{h}_k$ , its effective channel vector<sup>4</sup> is defined as  $\mathbf{e}_k$ , and can be calculated by projecting  $\mathbf{h}_k$  onto the orthogonal complement of the subspace spanned by set  $\{\mathbf{e}_1, \dots, \mathbf{e}_{(|S|)}\}$ , being interpreted as the effective channel vectors of previously selected users [9], [11]:

$$\mathbf{e}_k = \mathbf{h}_k - \sum_{j=1}^{|S|} \frac{\mathbf{h}_k \mathbf{e}_{(j)}^*}{\|\mathbf{e}_{(j)}\|^2} \mathbf{e}_{(j)} \quad (14)$$

$$= \mathbf{h}_k \left( I - \sum_{j=1}^{|S|} \frac{\mathbf{e}_{(j)}^* \mathbf{e}_{(j)}}{\|\mathbf{e}_{(j)}\|^2} \right) \quad (15)$$

$$= \mathbf{h}_k Q(S) \quad (16)$$

where  $I$  is the identity matrix and  $Q(S)$  is a projection matrix. Then, the ECG of user  $k$  is denoted as  $\|\mathbf{e}_k\|^2$ .

Specifically, for OFDM systems, which divide the bandwidth into orthogonal subcarriers and treat each of the subcarriers as an independent narrowband channel, the ECG should be averaged over all subcarriers<sup>5</sup>. Let  $\mathbf{e}_k[c]$  be the ECG on subcarrier  $c$ , then we have

$$\|\mathbf{e}_k\|^2 = \frac{1}{N_c} \sum_{c=1}^{N_c} \|\mathbf{e}_k[c]\|^2 \quad (17)$$

where  $N_c$  is the total number of subcarriers.

---

<sup>4</sup> The effective channel vector of the first user is the same as its channel vector,  $\mathbf{e}_{(1)} = \mathbf{h}_{(1)}$ .

<sup>5</sup> In this work, we do not consider a per-subcarrier decision. We leave this issue to our future work.

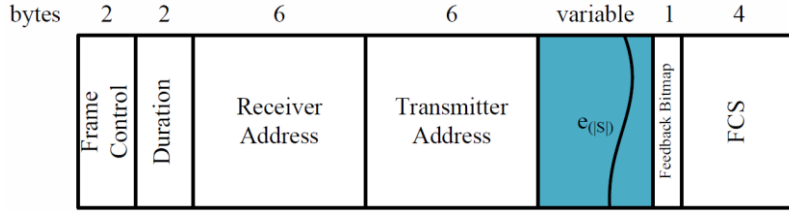


Figure 3.3: **The change in use of the polling frame.** When using a channel hint, polling is no longer destined for only one target user, so the receiver address should be the broadcast address. The length of the channel hint varies depending on the quantization level.

## 2) Channel Hint Design

There are many ways to implement a channel hint. In a simple way, the AP can use a projection matrix as a channel hint. Although this simplifies the computation burden at the receiver, it always consumes significant bits for representing an  $M \times M$  matrix per each subcarrier (the maximum required number of appended channel vectors is  $(M - 1)MN_c$ ). By contrast, if we use the received CSI as a channel hint, then this will increase the receiver side computing complexity.

Our proposed design exploits a compromise between the above two schemes. Every time the AP receives a CSI report, it computes the effective channel vector from that CSI and uses it as a channel hint. Hence, in every poll frame the effective channel vector of the last scheduled user will be included. On the receiver side, users should maintain the value of summation in (15) during the feedback phase so that they can update a projection matrix and obtain their ECGs much faster.

We illustrate the poll frame structure used in 802.11ac+ in Figure 3.3. As shown in the figure, the poll frame requires only one additional channel vector, and thus the maximum number of appended vectors is  $(M - 1)N_c$ . Since in each selection round, each user and the AP require one  $(1 \times M) \times (M \times M)$  vector-matrix multiplication per subcarrier, we conclude that the computational complexity is  $C(M - 1)N_c$ , where  $C$  is a computational cost corresponding to one vector-matrix multiplication.

One may claim that polling additional  $(M - 1)$  users and taking CSI from them for user selection would be more effective than using  $(M - 1)$  channel hints. However, this may have a large feedback overhead, which brings MU-MIMO performance degradation. To clarify this, we compare the overhead of two schemes. Let

$\tau_{ECV}$ ,  $\tau_{POLL}$  and  $\tau_{CSI}$  be the transmission time of sending one effective channel vector in channel hint, one polling frame and one CSI feedback, respectively. Note that  $\tau_{ECV} + \tau_{POLL}$  is the transmission time for one channel hint, and  $\tau_{ECV} < \tau_{POLL}$ . First, in 802.11ac+, the AP will send maximum  $(M - 1)$  channel hints, and maximum  $M$  users will report their CSI reports. Then, we have the following:

$$\begin{aligned} overhead_{802.11ac+} &= \tau_{POLL} + (M - 1)(\tau_{ECV} + \tau_{POLL}) + M\tau_{CSI} + 2M\tau_{SIFS} \\ &\quad + (M - 1)\tau_{contention} \end{aligned} \quad (18)$$

$$\begin{aligned} &= M\tau_{POLL} + (M - 1)\tau_{ECV} + M\tau_{CSI} + 2M\tau_{SIFS} \\ &\quad + (M - 1)\tau_{contention} \end{aligned} \quad (19)$$

where  $\tau_{SIFS}$  stands for 802.11 Short InterFrame Space (SIFS), which is  $16\mu\text{s}$  in 802.11ac [2]. Here,  $\tau_{contention}$  is an overhead from the feedback contention of 802.11ac+, which will be discussed in the next section. As will be shown in Figure 3.5 (b) later, the maximum value of a  $\tau_{contention}$  is about  $2\tau_{SIFS}$  (four slots per each round).

Now, assume that the AP polls  $(M - 1)$  users after taking CSI from  $M$  users. In other words, the AP takes CSI from total  $(2M - 1)$  users for user selection. Then, we have the following:

$$overhead_{add} = (2M - 1)(\tau_{POLL} + \tau_{CSI}) + 2(2M - 1)\tau_{SIFS} \quad (20)$$

$$= (2M - 1)\tau_{POLL} + (2M - 1)\tau_{CSI} + 2(2M - 1)\tau_{SIFS}. \quad (21)$$

Even if taking  $\tau_{ECV} \cong \tau_{CSI}$ ,  $overhead_{802.11ac+} < overhead_{add}$ . Also, since the AP has no choice but to poll users randomly in this scheme, the user selection gain could be small, as shown in the case of  $M = 4$  in Figure 1.2.

When sending a channel hint, since the AP may not intend all users to participate in the feedback procedure, it requires a method to notify only some dedicated users. For example, an AP should prevent users from sending CSI reports when it has no frames for them. For target user notification, we utilize the legacy Null Data Packet Announcement (NDPA) frame, by using fewer bits for the Association ID in the STA Info field and mapping bit positions to each user ID. This method maintains the same frame format, and thus is efficient. A detailed description of the notification is out of the scope of this thesis.

Table 3.1: An Example of the Thresholds

CR	$(w_s, w_c, w_t)$	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$
1	(1,1,1)	3.667	3.543	3.453	3.371	3.280
	(0.4,0.4,0.2)	3.680	3.558	3.472	3.396	3.318
2	(1,1,1)	2.541	2.436	2.360	2.291	2.215
	(0.4,0.4,0.2)	2.552	2.449	2.376	2.312	2.246
3	(1,1,1)	1.384	1.304	1.247	1.196	1.140
	(0.4,0.4,0.2)	1.392	1.314	1.259	1.212	1.163

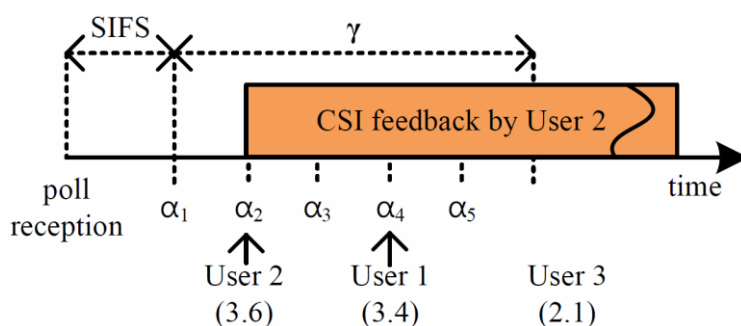


Figure 3.4: **Example of the active CSI feedback with  $\gamma$  of five slots.** Slot thresholds are given in Table 3.1. User 2 wins the contention because its ECG is higher than that of the others. Other users are notified of User 2's transmission and wait for the next round.

### 3.4.3 Active CSI Feedback

#### 1) Feedback Contention

In 802.11ac+, only the user with the highest ECG should respond to the poll. However, this is a challenging task since users cannot know the ECG of the others in a distributed manner. If two users, who might think their gains are the best, feed back their CSI reports simultaneously, a frame collision will occur. Without any coordination between users, they should get opportunities to send feedback through a contention like the 802.11 Distributed Coordinate Function (DCF). We call this a *feedback contention* in this chapter.

In order to resolve the feedback contention, we adopt a delayed transmission approach. In this approach, users delay their feedback transmission regarding their

ECGs, such that a user with higher value can access the feedback opportunity faster. To realize this, we first apply slotted time in our scheme, where the slot duration is fixed and each slot has a pre-defined threshold. We refer to this threshold as a *slot threshold*. A user is assigned to the specific slot according to its ECG and is allowed to transmit feedback only at the slot. If a user senses a feedback transmission earlier than its slot, it gives up and waits until the next poll.

Since users should listen to the medium until they obtain a transmission opportunity, the minimum duration of the slot requires at least one slot time of 802.11 (e.g.,  $4\mu\text{s}$  for CCA +  $5\mu\text{s}$  for RX/TX turnaround). To limit the expense of delayed feedback, an AP uses the timeout threshold ( $\gamma$ ). If the AP does not receive any feedback reports before  $\gamma$ , then it finishes the scheduling. When feedback collision or timeout occurs in the first contention round, the scheduling may be stopped with only one user (i.e., the first user) remaining in  $S$ . In this case, the AP just sends a frame to that user in diversity mode, which guarantees the minimal level of system performance.

Figure 3.4 shows an example of an active feedback scheme with three users and five feedback slots for the first contention round. The associated slot thresholds are given in Table 3.1, which will be explained in the following section. Assume that User 2 and User 1 can transmit CSI feedback in the second and fourth slots, respectively, while User 3 cannot get an opportunity due to the low gain. User 2 sends the CSI feedback in the second slot. The others listen to the feedback transmission and wait for the next contention round.

## 2) Slot Threshold Optimization

The performance of the active feedback mainly relies on how to select the thresholds:  $\gamma$  and slot thresholds. As described before, active feedback fails when a collision or timeout occurs; however, their impacts are different from each other. The loss from collision is much bigger than that from timeout, due to the relatively long time wasted in the collision. Thus, we first fix  $\gamma$  and find optimal values of the slot thresholds for each contention round. Now, we describe how to determine the thresholds as below.

Provided that a random channel vector on subcarrier  $c$ ,  $\mathbf{h}_k[\mathbf{c}]$ , is a zero-mean circularly symmetric complex Gaussian random variable, from (17),  $\|\mathbf{e}_k\|^2$  is a

Gamma-distributed random variable with  $N_c L$  (shape parameter) and  $\frac{1}{N_c}$  (scale parameter), where  $L = \text{rank}(Q(S))$ . Formally, for a constant  $\alpha$ ,  $\|\mathbf{e}_k\|^2$  satisfies

$$\text{P}\{\|\mathbf{e}_k\|^2 \leq \alpha\} = F(\alpha; N_c L, \frac{1}{N_c}) \quad (22)$$

where  $F(\alpha; N_c L, \frac{1}{N_c})$  is the Cumulative Distribution Function (CDF) of the Gamma distribution for  $\alpha$ ,  $N_c$  and  $L$ . For fixed  $N_c$  and  $L$ , we use  $F(\alpha)$  and  $F(\alpha; N_c L, \frac{1}{N_c})$  in the same sense. Note that  $L$  varies every contention round.

For a timeout  $\gamma$  ( $= G$  slots) in a certain contention round, let  $\alpha_1, \alpha_2, \dots, \alpha_G$  be the threshold corresponding to each slot, where  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_G \geq 0$ . We assume that  $\gamma$  is fixed over all contention rounds. We define the probability of successful feedback as the probability that only one user will send a CSI report in  $\gamma$ . Then, it can be calculated by summing up all success probabilities over all time slots:

$$p^{suc} = \sum_{g=1}^G p_g^{suc} \quad (23)$$

$$= K' \{1 - F(\alpha_1)F(\alpha_1)^{(K'-1)}\} + K' \{F(\alpha_1) - F(\alpha_2)F(\alpha_2)^{(K'-1)}\} + \dots + K' \{F(\alpha_{G-1}) - F(\alpha_G)F(\alpha_G)^{(K'-1)}\} \quad (24)$$

where  $K'$  is the number of contending users in the contention round. We note that  $K' \leq K - 1$  because the first user is already selected by AP, and  $K'$  can be estimated by using the target user notification.

The timeout and collision probabilities are defined as follows:

$$p^{tout} = F(\alpha_G)^{K'} \quad (25)$$

$$p^{col} = 1 - p^{suc} - p^{tout}. \quad (26)$$

Then, our objective is to find a set of thresholds as follows:

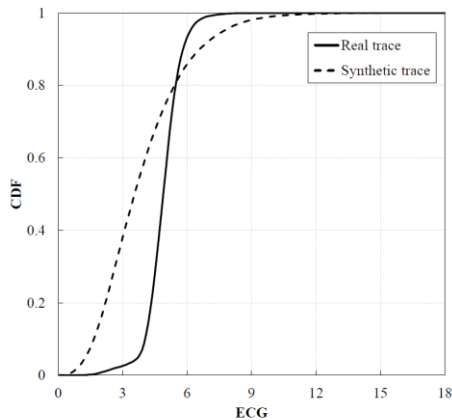
$$\underset{\alpha_1, \dots, \alpha_G}{\text{argmax}} w_s p^{suc} - w_c p^{col} - w_t p^{tout} \quad (27)$$

where  $w_s$ ,  $w_c$  and  $w_t$  are weights for the probabilities of success, collision and timeout, respectively.

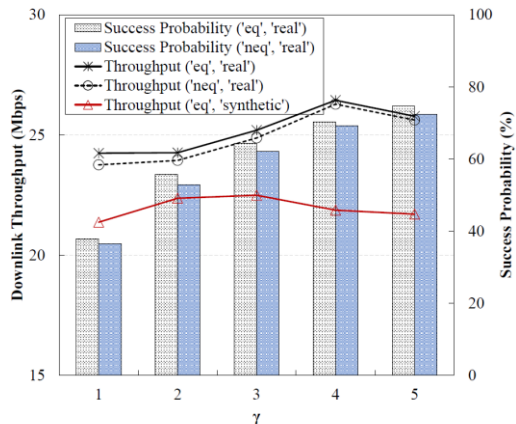


The optimization problem described in (27) is non-convex and is solved using nonlinear optimization technique. Note that it only needs to be solved whenever the network configuration  $(N_c, M, K)$  is changed or we can construct a set of solutions offline. Once a set of thresholds has been constructed, users can refer to them at any time.

Table 3.1 shows an example of the thresholds in the case of  $\gamma = 5$ ,  $M = 4$ , and  $N_c = 30$ , where CR stands for contention round. Recall that there are maximum  $(M - 1)$  contention rounds and each round consists of  $\gamma$  time slots. From the table, we can see that the thresholds decrease during the course of a contention round, because successful feedback reduces  $L$  one by one, and hence decreases the DoF of the Chi-Square distribution. In addition, the reduced number of contending users also results in lower thresholds in further contention rounds. The thresholds of nonequal weights are slightly bigger than those of equal weights. In this case, the probability of success as well as the probability of collision decreases.



(a) ECG distributions.



(b) Performance comparison with respect to  $\gamma$ .

**Figure 3.5: Performance of the active CSI feedback.** (a) We use two different CSI traces (real and synthetic traces). The channel coefficients of the synthetic trace follow the Rayleigh distribution explained in Section 3.3.1. (b) We use weights of (0.4,0.4,0.2) as the non-equal weights cases. The results for ('neq', 'synthetic') are omitted, because they show a similar pattern as those for 'real'.

We compare the downlink throughput and success feedback probability of 802.11ac+ as a function of  $\gamma$  in Figure 3.5. In particular, to observe the impact of the threshold optimization according to channel distributions, we use two different channel distributions. Figure 3.5 (a) shows the ECG distributions of two traces used in this simulation: 'real trace' and 'synthetic trace (Rayleigh)'. Note that the slot thresholds are computed from the real trace. As shown in Figure 3.5 (b), a throughput loss (maximum 4Mbps when  $\gamma = 4$ ) happens in the synthetic trace scenario. This result is predictable: the thresholds are optimized only for the real trace. Even though the performance loss might seem small in this result, using a more exact channel model will improve the robustness of the proposed scheme. Note that in this section we provide an analysis of the Rayleigh fading case only, but we can apply the same technique for different distributions, e.g., Rician, by changing (22) in the optimization.

The graph also demonstrates that non-equal weights reduce both the feedback success probability and the collision probability (omitted here), as expected, thereby decreasing the scheduling gain. As a result, the throughput of non-equal weights is lower than that of the equal weights over all cases. Additionally, we can see that too large a  $\gamma$  decreases the effective channel gain while too small a  $\gamma$  decreases the user

diversity gain. For the real trace scenario, both downlink throughputs increase as  $\gamma$  reaches four; after that they decrease (for the synthetic trace, maximum downlink throughput is obtained when  $\gamma$  is three).

## 3.5 Fair Scheduling

In this section we develop two fair scheduling approaches of the 802.11ac+ based on the Round-Robin and Proportional-Fair algorithms. Two main changes are adopted in the fair scheduling protocols to maintain the property of the 802.11ac+ user scheduling. The first change is to make use of a simple fair queue in the AP of both RR-11ac+ and PF-11ac+. In particular, the AP selects the first user by using the following:

$$\operatorname{argmax}_{k \in A} \frac{1}{\mathcal{R}_k} \quad (28)$$

where  $\mathcal{R}_k$  is the average data rate of user  $k$  and  $A$  is a target user set. For the remaining users, each fair scheduling protocol uses a different scheduling strategy, but the main idea is the same as that of 802.11ac+. We describe the details of RR-11ac+ and PF-11ac+ in the next subsections.

### 3.5.1 RR-11ac+

This is the simplest way to obtain the fairness that gives users equal transmission opportunities. Most Round-Robin schedulers use the average data rates of all users and they can be monitored at the scheduler, which is very easy to implement.

In RR-11ac+, we use the same metric to schedule users, but a simple change is made for multi-user transmission: before the active CSI feedback, an AP limits the contending users according to their average data rates. Specifically, among remaining users in  $A$ , the AP selects top  $(M - 1 + o)$  users in ascending order of average data rates such that users who have been served less have more feedback opportunities. If the non-negative parameter  $o$  is set to 0, then RR-11ac+ is reduced to the 802.11ac with the legacy Round-Robin scheduling; however, it still considers the ECG of users. For example, if one's ECG is too low so joining it into  $S$  is no longer helpful for multi-user transmission, then it will be discarded.

Table 3.2: Comparison between Three Schemes

	Size of target user set	First user selection	Metric for $n(> 1)$ th user selection
802.11ac+	$K$	Random or $\operatorname{argmax} \ \mathbf{e}_k\ ^2$	$\ \mathbf{e}_k\ ^2$
RR-11ac+	$M - 1 + o$	$\operatorname{argmax} \frac{1}{\mathcal{R}_k}$	$\frac{T_k}{\mathcal{R}_k}$
PF-11ac+	$K$		$\frac{T_k}{\mathcal{R}_k}$

### 3.5.2 PF-11ac+

The proportional fairness maintains a balance between maximizing the sum rate and allowing all users at least a minimal level of service. Basically, both the original Proportional-Fair algorithms and PF-11ac+ use the following criterion to select users:

$$\frac{T_k}{\mathcal{R}_k} \quad (29)$$

where  $T_k$  is the current available data rate of user  $k$ .

The difficulty of legacy Proportional-Fair scheduling is that the scheduler cannot know the exact data rates of users ( $T_k$ ) when the scheduling decision has to be made. However, the active feedback of 802.11ac+ provides a simple implementation of the Proportional-Fair scheduling since users can actively join the scheduling decision. In particular, they can estimate  $T_k$  by using their ECGs (i.e.,  $T_k \sim \log(\|\mathbf{e}_k\|^2)$ ). Note that in PF-11ac+ users should maintain their average data rates, and the slot thresholds should be redefined according to (29) by adopting the technique in Section 3.4.

### 3.5.3 Summary

Table 3.2 shows the comparison between 802.11ac+, RR-11ac+ and PF-11ac+. As shown in the table, there are notable differences in three aspects: target user set, first user selection and  $n$ th user selection. First, in 802.11ac+ and PF-11ac+ all users are considered for scheduling, while in RR-11ac+ the number of target users is bounded to  $(M - 1 + o)$ , as explained earlier. Second, in 802.11ac+ basically the first user is selected randomly according to AP's queuing policy, while in RR-11ac+ and PF-11ac+, the user with maximum utility (28) will be chosen. To maximize the sum

capacity, the user with the largest ECG will be selected as a first user, similar to SUS [9]. In using this method, one possible concern is that optimally selecting the first user is difficult since no CSI is available at the time of user selection. To handle this issue, we propose referring to the statistics of previous channel status of the users, for the first user selection. This method should work well when transmissions of the AP happen in channel coherent time, which typically ranges from  $15ms$  to  $100ms$  [6], [58]. Lastly, for  $n$ th user selection, 802.11ac+ uses ECG as a selection metric, while the fair scheduling protocols use (29) as a selection metric.

Table 3.3: **Default Simulation Parameters**

Parameter	Value
Downlink (Uplink) traffic intensity	5 (0.5) Mbps
$K$	15
The number of polls used in SUS	10
$M$	4
SNR	16.7 dB
$\gamma$	4
$\sigma$	4

## 3.6 Performance Evaluation

In this section, simulation results for the throughput and fairness performance of the proposed schemes are presented.

### 3.6.1 Setting

We implement 802.11ac, SUS, 802.11ac+, RR-11ac+ and PF-11ac+ on the MATLAB simulator. To conduct high-fidelity emulation of real-world settings, we use the 802.11n data traces provided by the authors of [59]. The traces contain per-subcarrier (30 subcarriers for 20MHz) CSI and SNR readings (ranging from 4dB to 43dB) for 18 users. By using the traces, we reconfigure 25 users and a maximum of four AP antennas in our simulations.

Except for the set of fair scheduling protocol simulations, we assume that all users have the same average SNR of 16.7 dB (according to the trace). We set the simulation parameters to the default values in Table 3.3 and 802.11ac. The AP and users generate traffic for each other according to their average sending rates (traffic intensity). We measure the average aggregate throughput of the downlink, uplink and system. Note that the system throughput is defined as the sum of downlink and uplink throughput. We also assume that all protocols used in the simulation enable a packet aggregation scheme.

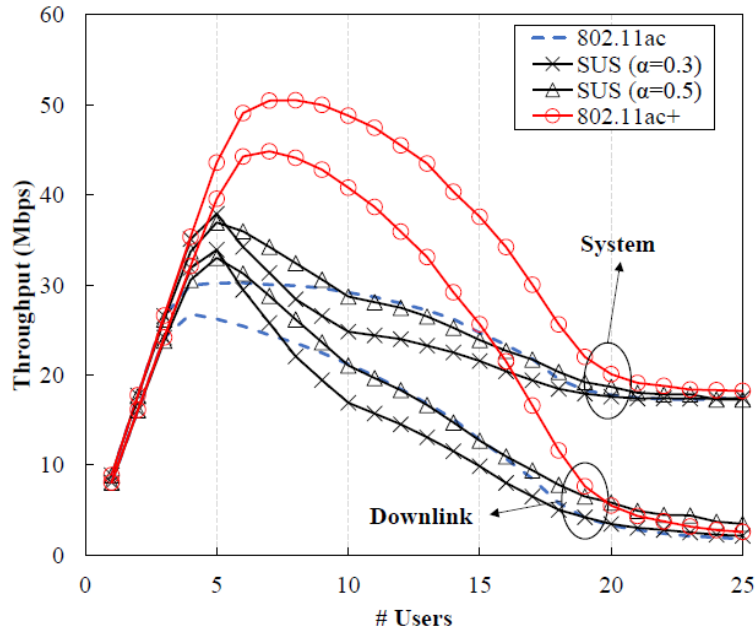


Figure 3.6: **Throughput according to the number of users.** Due to dominant uplink traffic and heavy contention, the network throughput decreases with the number of users. We note that  $\alpha$  used in SUS controls the trade-off between diversity gain and effective channel gain [9].

### 3.6.2 802.11ac+ Performance

#### 1) Impact of $K$

The user diversity will be more effective as the number of users to be considered for the scheduling increases; however, the significant CSI feedback overhead may limit the gain. In order to examine the performance of 802.11ac+, we illustrate the downlink and system throughput of three protocols according to the number of users in Figure 3.6. Both throughputs increase until the number of users reaches a certain point, after which they start to decrease. 802.11ac+ shows significant throughput gain over the other protocols from the high channel utilization via its user selection scheme. As a result, it achieves the maximum downlink (system) throughput gain of 100% (69%) and 168% (97%) over 802.11ac and SUS, respectively. On the other hand, SUS shows a similar or worse performance compared to 802.11ac except for a moderate number of users.



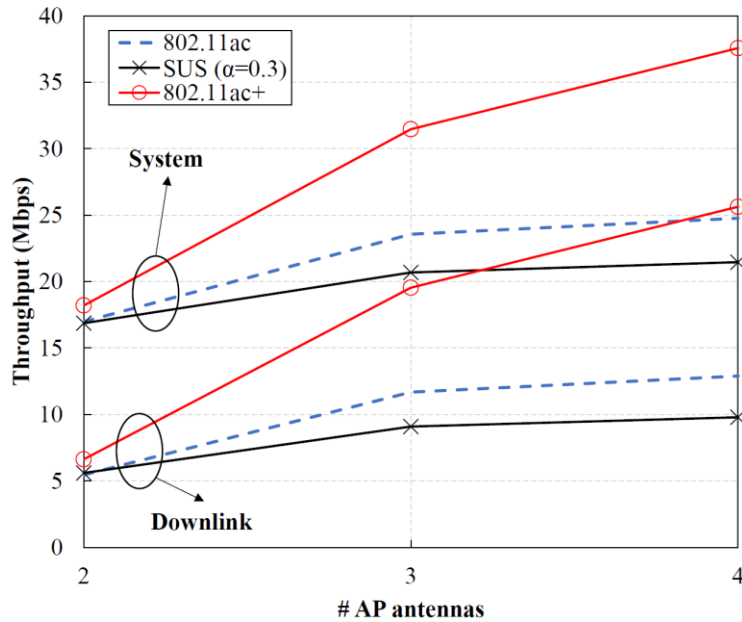


Figure 3.7: **Throughput according to the number of AP antennas.** The large number of AP antennas increases the size of the CSI feedback frame. Thus, requesting many users' CSI for scheduling may limit the scheduling gain significantly, as shown in the result of SUS. However, 802.11ac+ can obtain higher throughput gain over both protocols due to the limited feedback.

## 2) Impact of $M$

More AP antennas provide better spatial multiplexing gain while increasing the CSI overhead because a large number of bits are required for representing CSI. In this simulation, we investigate and compare the performance according to the number of AP antennas, as shown in Figure 3.7. As expected, the throughput of protocols increases with the number of AP antennas. However, the CSI overhead limits the performance of SUS while 802.11ac+ can obtain much higher throughput gain due to small CSI overhead. As a result, it achieves the maximum downlink (system) throughput gain of 98% (51%) and 161% (75%) over 802.11ac and SUS, respectively.

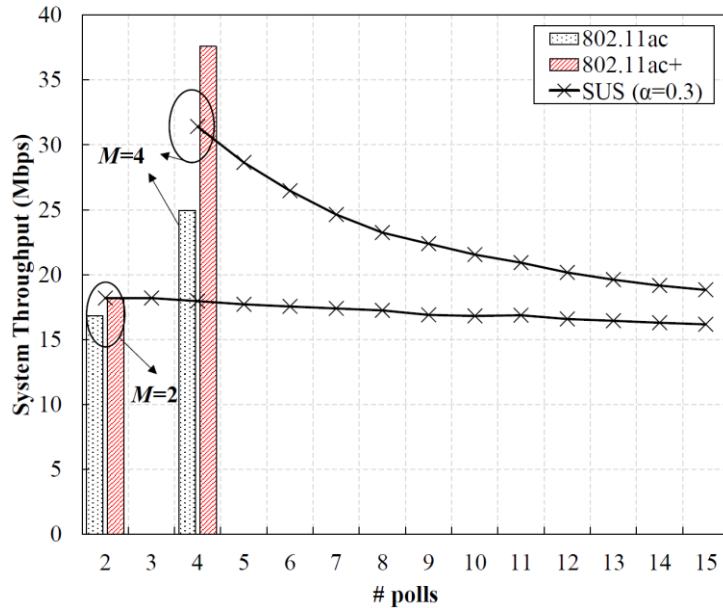


Figure 3.8: **System throughput according to the number of polls.** 802.11ac+ obtains higher throughput gain from the user scheduling with a fixed number of polls, while in the case of SUS, the performance degrades as the number of polls increases.

### 3) Impact of the Number of Polls

To investigate the user scheduling gain affected by the CSI overhead, we compare the three system throughput results according to the number of polls in Figure 3.8. As expected, SUS suffers from the CSI overhead as the number of polls increases. As a result, SUS shows poor throughput performance compared to 802.11ac after the number of polling frames exceeds seven in the case of  $M = 4$ . In contrast to SUS, 802.11ac+ outperforms other protocols by using far less CSI feedback while fully harnessing the user scheduling gain. As a result, in the case of  $M = 4$  and four polling frames, its throughput gains over 802.11ac and SUS are 51% and 20%, respectively.

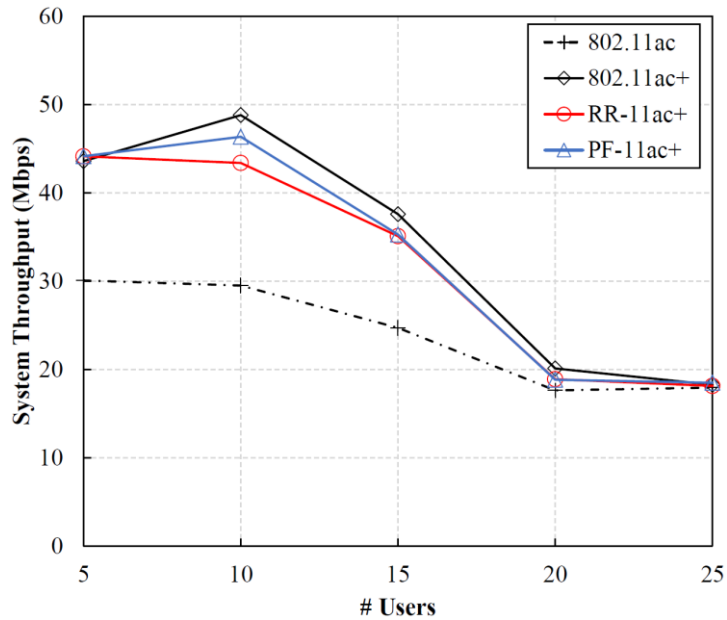


Figure 3.9: **System throughput comparison of 802.11ac, 802.11ac+ and two fair scheduling protocols.** 802.11ac+, RR-11ac+, and PF-11ac+ show a similar performance under the same average user SNR environment due to the fact that they consider the effective channel gain.

### 3.6.3 Fair Scheduling Protocol Performance

#### 1) System Throughput

We compare the system throughput of RR-11ac+ and PF-11ac+ with 802.11ac and 802.11ac+ in Figure 3.9. As shown in the graph, 802.11ac+ and its fair approaches show similar throughputs, although they use different user selection criteria from each other. In particular, the system throughputs of RR-11ac+ and PF-11ac+ are much bigger than that of 802.11ac, since the user scheduling adopted in them still considers the effective channel gain, unlike 802.11ac. Recall that all users experience similar average SNRs in this simulation.

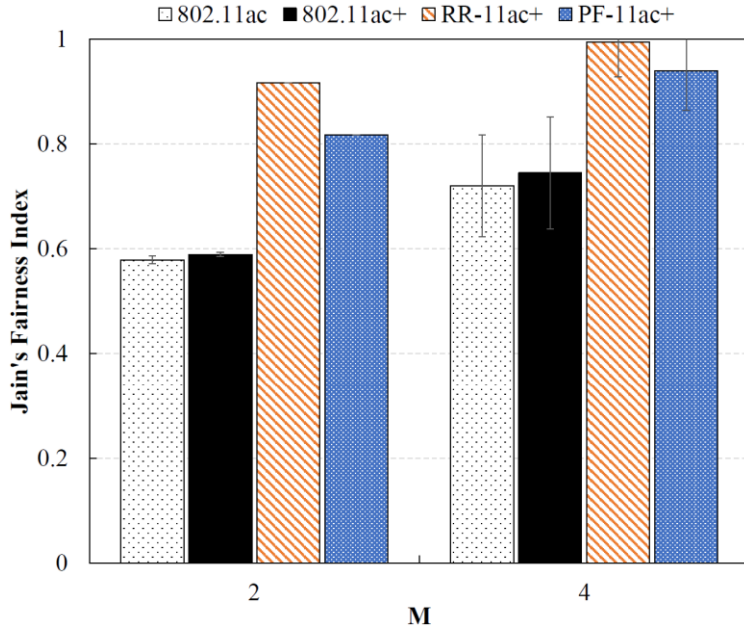


Figure 3.10: **Jain's Fairness Index according to  $M$ .** RR-11ac+ and PF-11ac+ can achieve high throughput fairness compared to 802.11ac and 802.11ac+.

## 2) Fairness

To evaluate the fairness performance of RR-11ac+ and PF-11ac+, users are subjected to different average SNRs from 4dB to 43dB, and a user with a higher ID has a bigger SNR than one with a lower ID, unlike in the previous set of simulations.

Figure 3.10 illustrates Jain's Fairness Index of downlink throughput of four protocols. From the result, we can see that the MU-MIMO transmission with more antennas gives better fairness because it can serve more users at once. Over all cases, RR-11ac+ shows the best fairness performance, followed by PF-11ac+. In particular, RR-11ac+ with  $M = 4$  achieves close-to-perfect fairness among users. In the meantime, 802.11ac and 802.11ac+ bring low fairness levels compared with the two fair scheduling protocols. The performance of 802.11ac+ is similar to or higher than that of 802.11ac, and yet the sum rate of 802.11ac+ is much higher than that of 802.11ac.

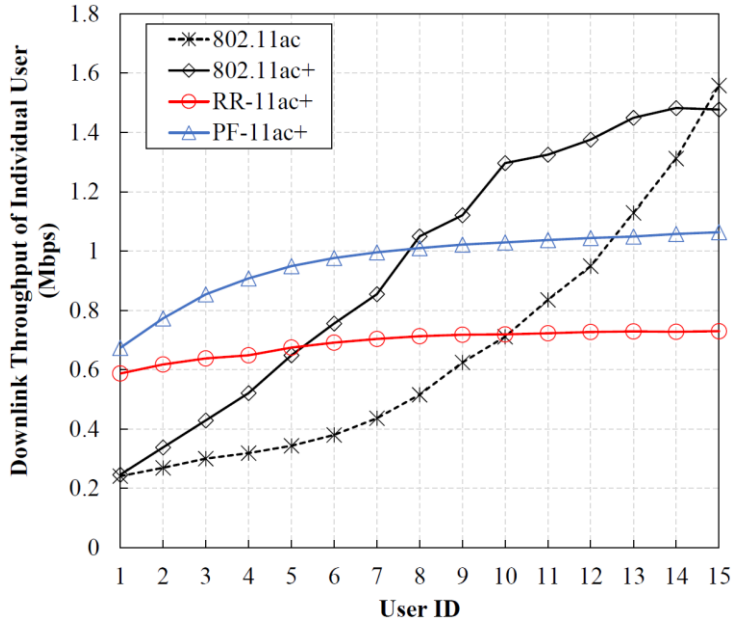


Figure 3.11: **Downlink throughput comparison of 802.11ac, 802.11ac+, RR-11ac+ and PF-11ac+.** RR-11ac+ and PF-ac+ achieve fairly similar throughputs over all users, though they experience different channel qualities.

Finally, we plot the downlink throughput that each user attains under each protocol in Figure 3.11. From the figure, we can clearly see that RR-11ac+ and PF-11ac+ achieve fairly similar throughputs over all users except some users with low SNRs. In addition, in this result, PF-11ac+ shows better aggregate downlink throughput than 802.11ac+, since users in 802.11ac+ have different chances of being scheduled regarding only their ECGs: the user scheduling of 802.11ac+ only favors a user to enhance the sum rate. As a result, there exist some users that have low throughput in 802.11ac+. Surprisingly, in the case of 802.11ac, this phenomenon is more obvious; more users suffer from starvation and a few users with high SNRs enjoy the high throughput. The main reason is that since the user scheduling in 802.11ac+ cares about the channel orthogonality between users, the probability of creating a low-benefit scheduling group is lowered.

### **3.7 Conclusion**

In this work, we propose a new MU-MIMO MAC protocol, 802.11ac+, which obtains significant user scheduling gain with a far smaller amount of CSI feedback by exploiting channel hint-based polling and active CSI feedback. Trace-driven MATLAB simulation results show that 802.11ac+ achieves downlink throughput gains of up to 100% and 168% over 802.11ac and SUS-based MAC protocols. Additionally, two fair scheduling protocols of 802.11ac+ give close-to-perfect user fairness even when the users experience different channel qualities.

# CHAPTER IV

## Distributed Frequency Domain User Selection

### 4.1 Introduction

In this chapter, we present a new user selection protocol called DiFuse (Distributed frequency domain user selection) that uses the capacity gain as a scheduling metric in user selection. The key mechanism of DiFuse is to greedily select a user at each iteration which yields maximum positive increment to the sum-capacity of the network. Given a user set, the capacity gain of a new user is defined as the increment in network capacity achieved by including the new user to the user set. DiFuse, as its name suggests, makes the scheduling decisions in a distributed manner. Each user cleverly computes its expected sum-capacity gain by overhearing the CSI feedback transmissions from other users. Then each user sends its sum-capacity gain in a simplified format called a Selection REQuest (SREQ); the user marks on one particular subcarrier of an OFDM symbol depending on the value of sum-capacity gain. The users concurrently transmit the SREQs via distributed feedback contention,

which is devised to effectively reduce feedback contention via dynamic threshold design and frequency domain contention [36], [60], [61]. The AP receives the SREQs and collectively uses them to select the user that gives the maximum positive increment to the sum-capacity of the network, then polls the user for the actual CSI transmission.

We implement the DiFuse prototype on the USRP N210 and GNURadio [13], and conduct testbed experiments and trace-driven emulations. The results show that DiFuse obtains higher throughput compared to conventional schemes such as SUS [9] and OPUS [12], as well as 802.11ac [2]. Further, DiFuse gives a better degree of proportional fairness over other schemes, especially when users experience different channel qualities.

The remainder of this chapter is organized as follows. We give some background of MU-MIMO and user selection in Section 4.2. We discuss the DiFuse mechanism in Section 4.3. Section 4.4 shows the performance evaluation and we finally conclude our chapter in Section 4.5.

## 4.2 Motivation

For optimal user selection, we should consider all possible user groups, compare their capacities, and choose one group which gives the highest capacity. However, such a naïve and exhaustive search over the entire user set (i.e.,  $\sum_{m=1}^M \binom{K}{m}$ ) obviously results in very high feedback overhead as well as computation, especially when  $K$  is very large. To reduce the overhead, many schemes adopt the incremental user selection approach: the AP or BS (Base Station) incrementally chooses a user by employing a user selection metric in each iteration, e.g., channel strength, orthogonality (angle), (projected) norm and capacity gain.



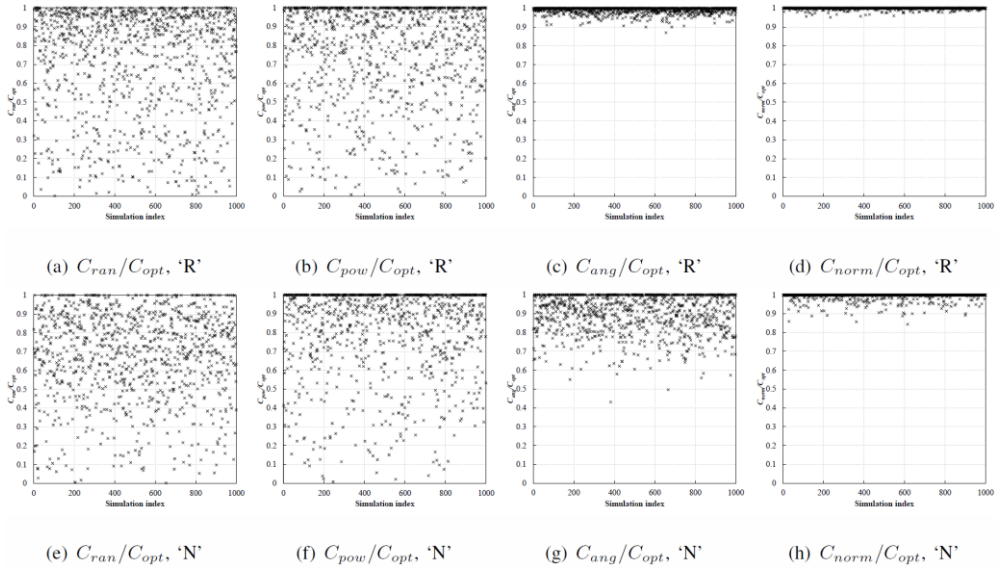


Figure 4.1: **Normalized capacity gain of each selection metric.** We set  $M = 2, K = 20$  and use two traces, real trace ('R') and synthetic trace ('N').

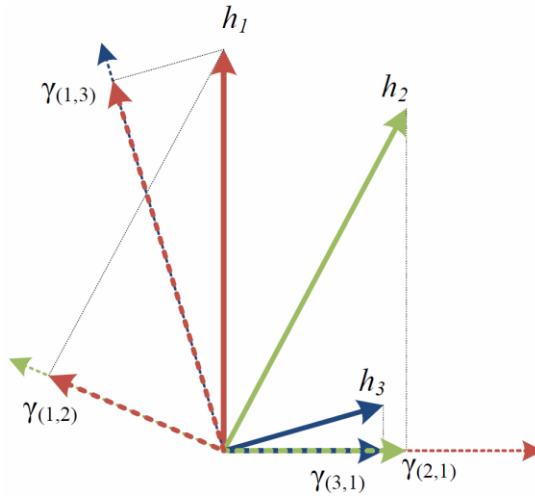


Figure 4.2: **An illustrative example of the projected norm based scheme when  $M = 2$  and  $K = 3$ .** We assume that User 1 was already selected.  $\gamma_{(a,b)}$  denotes the channel of user  $a$  projected to the orthogonal subspace of channel of user  $b$ . User 2 will be selected if we select the user with the largest projected norm, i.e.,  $\gamma_{(2,1)} > \gamma_{(3,1)}$ . However, the actual sum-capacity becomes bigger when User 3 is selected instead of User 2, i.e.,  $\log \gamma_{(1,2)} + \log \gamma_{(2,1)} < \log \gamma_{(1,3)} + \log \gamma_{(3,1)}$ .

Figure 4.1 compares the normalized capacity gain of the four selection metrics (random, maximum power, maximum angle, maximum projected norm), computed over that of the optimal selection. We select one user randomly as the first user and then, choose the other users according to the selection metric. The same first user is chosen for each metric, for fair comparison. We set  $M$  as 2 and  $K$  as 20. For the optimal selection, we consider the capacity gain among all possible user groups. We use two type of traces as input: real trace ('R') and synthetic trace ('N').

As shown in the result, in random and max-power selection schemes, only about 15% and 20% result the same as the optimal, respectively. The max-angle metric is better by showing 27% to 46% of the optimal. Although the projected norm delivers the best performance (80% result the same as the optimal), it gives around 20% of non-optimal sum capacity. Such a performance loss happens because sometimes a user with a high power gain but small orthogonality could be selected. Such a small orthogonality may induce significant power loss to the previously selected user channels, as shown in the example in Figure 4.2. From this result, we conclude that the capacity gain is the most effective selection metric for the case of  $M = 2$ . Note that when  $M > 2$ , even the capacity gain cannot guarantee the optimal solution due to the nature of incremental user selection, but it still provides a better selection result than the projected norm, as will be shown later.

Unfortunately, exploiting the capacity gain as a selection metric is a challenging task. First, to compute the capacity gain, all CSI feedbacks from users should be sent prior to the selection, which significantly increases a MAC overhead. One alternative is to compute them in a distributed manner. In other words, we can let each user estimate its own capacity gain. To realize this, users should know the CSI of the previously selected users, and thus the AP may need to send them back to the remaining users, which brings a large amount of frame exchanges. Instead of explicitly sending the CSI feedbacks, the OPUS [12] performs orthogonality probing to realize distributed user selection. However, since the users do not have the CSI of others, the OPUS has no choice but to use the projected norm. Second, an effective contention mechanism should be supported for distributed user selection. A time-domain contention scheme adopted in OPUS is simple and intuitive, but the contention overhead is non-negligible, as mentioned earlier.

In summary, the main goal of this chapter is to propose a protocol that enables the capacity gain metric for user selection in a scalable manner and at the same time addresses the time-domain contention overhead issue. Our proposed scheme, DiFuse, meets both requirements by using overhearing-based CSI acquisition and frequency domain signaling. We elaborate the details of our scheme in the next section.

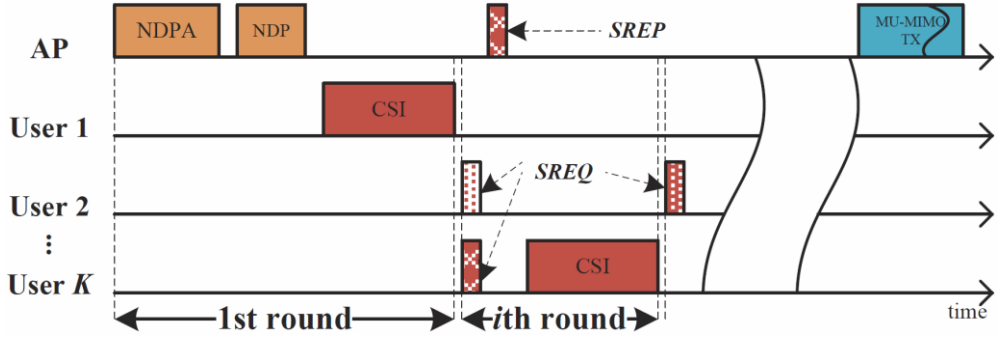


Figure 4.3: **Overview of DiFuse operation.** The poll and CSI feedback exchanges between the AP and users are conducted via SREQ/SREP based on frequency domain contention. DiFuse requires up to  $M$  polling and feedback frames to achieve the user selection gain.

## 4.3 Distributed Frequency Domain User Selection

### 4.3.1 Protocol Overview

Figure 4.3 illustrates the operation of DiFuse. Let us assume User 1 was first selected (we will explain the first user selection in the later subsection.). User 1 sends its CSI report to the AP, and all remaining users ( $k = 2, \dots, K$ ) overhear it<sup>6</sup> and compute the capacity gain by comparing the sum capacities of the two user groups: with and without itself. Then, the users concurrently send the gain value via SREQ to the AP, only if the gain is above 1. This implies that a user can become a candidate only if it gives a positive increment to the sum-capacity. Furthermore, since the users with gain values under 1 do not transmit the SREQ, the contention can be reduced. Say that User  $K$  has the highest capacity gain among all remaining users, i.e.,  $\frac{C(\{1,K\})}{C(\{1\})} > \frac{C(\{1,k\})}{C(\{1\})}, k \in \{1,2, \dots, K\} \setminus \{1, K\}$ . Then, in the second polling round, the AP polls User  $K$  via Selection REPLY (SREP) as the next user, and in turn, User  $K$  transmits the actual CSI feedback. DiFuse exploits the frequency domain contention to efficiently

<sup>6</sup> We assume explicit CSI feedback as in 802.11ac [2].

integrate the concurrent SREQ transmissions from the users. This process repeats while the remaining Degree of Freedoms (DoFs) are available.

### 4.3.2 Distributed Feedback Contention

We conduct the distributed feedback contention at the frequency domain. The capacity gain of each user is first mapped to a particular subcarrier of an OFDM symbol (setting a bit "1"), and then the users concurrently transmit the symbols via SREQ. The AP can detect these combined multiple SREQs by using the typical FFT. An SREQ lasts only for a few OFDM symbols, and so its overhead is much smaller than that of the time-domain contention [12]. Note that multiple OFDM symbols could be misaligned due to several reasons such as different propagation delay or switching delay, but the total misalignment has been shown to be tightly bounded [36], [60], [61]. As long as the misalignment is less than the Cyclic Prefix (CP), the AP can decode misaligned signals.

In making a SREQ, DiFuse uses a simple threshold based mapping scheme. We assume that for each polling round  $i$ , each data subcarrier  $n \in N$  has its virtual threshold  $E_{i,n}$  (we call this a slot threshold). For all  $i$  and  $n$ , all slot thresholds satisfy the following condition:

$$1 \leq E_i^{min} \leq E_{i,n} \leq E_i^{max} \quad (30)$$

$$E_{i,n} < E_{i,n+1} \quad (31)$$

, where  $E_i^{min}$  and  $E_i^{max}$  are the minimum and maximum slot threshold of round  $i$ , respectively. Recall that  $E_i^{min}$  cannot be less than 1.

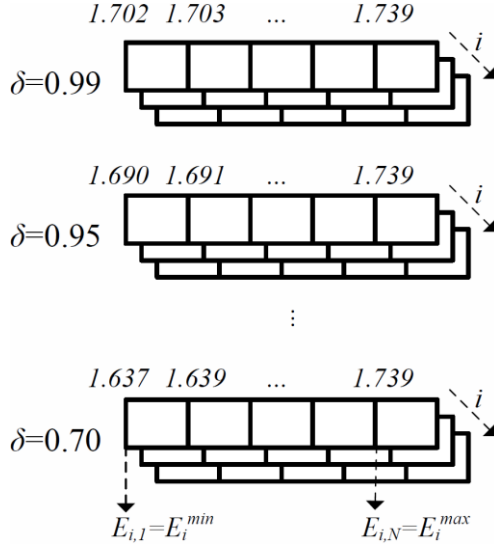


Figure 4.4: **The main concept of slot thresholds of DiFuse.** We generate several slot thresholds for idle case. Here  $\delta$  presents a seed parameter.

We illustrate an example of slot thresholds in Figure 4.4. There are several slot thresholds for each selecting round  $i$ , and when making SREQ, DiFuse users use one of them. In particular, at the beginning, users use slot thresholds with  $\delta = 0.99$ , and according to SREQ transmission results, users can use other threshold with the decreased  $\delta$ . The detailed description of  $\delta$  and the slot threshold is given in the next subsection. Given slot thresholds, a channel gain of User  $k$  ( $\lambda_{i,k}$ ) is mapped to  $n$ th subcarrier, if it satisfies the following condition:

$$E_{i,n} \leq \lambda_{i,k} < E_{i,n+1} \quad (32)$$

, where  $E_{i,|N|+1} = \infty$ .

Since the slot thresholds of round  $i$  are monotonic increasing with  $n$ , as shown in (30) and (31), a user with a higher capacity gain will activate a higher subcarrier index on SREQ. When users send SREQs, according to the capacity gain status of users, one of following cases may occur.

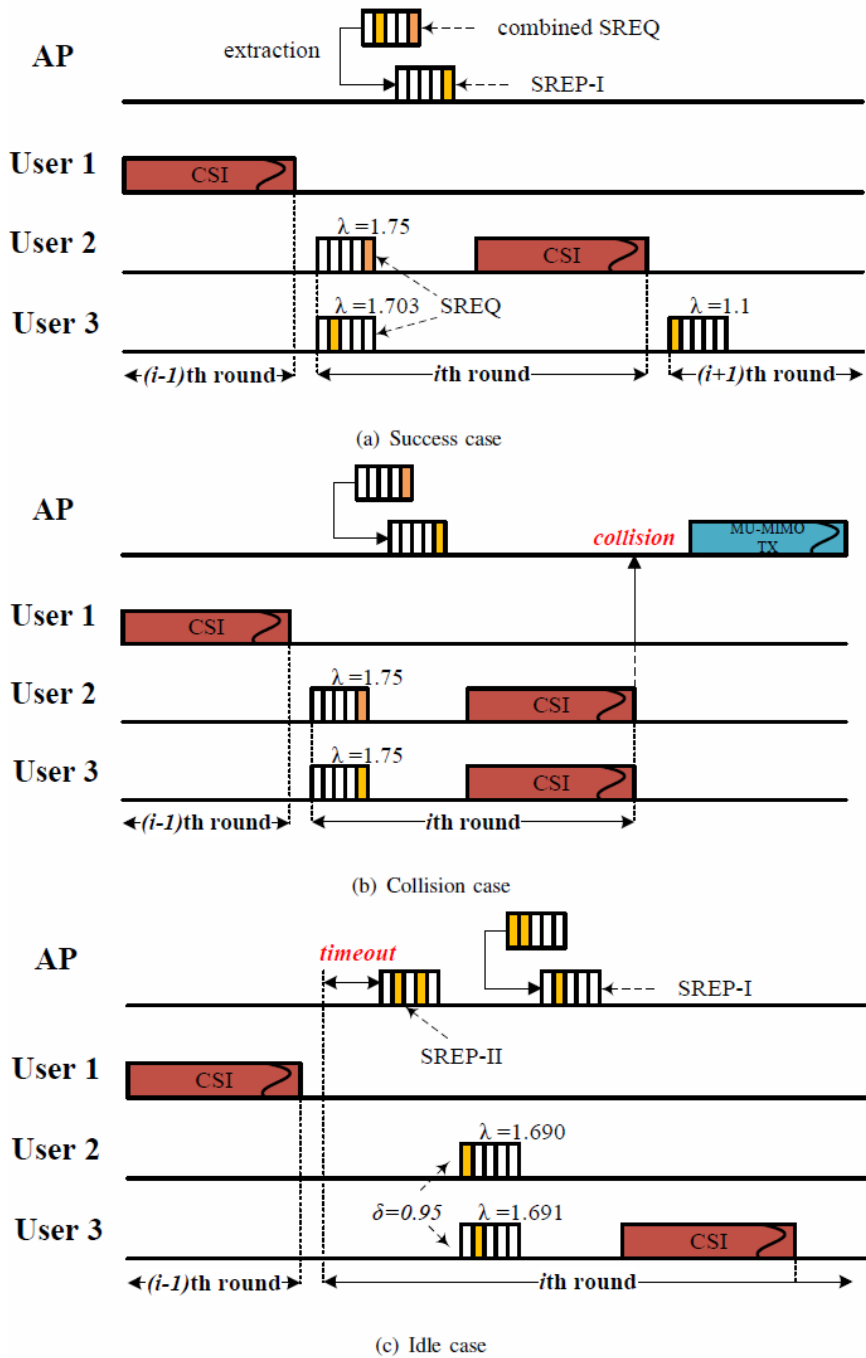


Figure 4.5: Three cases of SREQ transmissions.

- **Success case**

A success case happens only when a single user is associated with the highest activated subcarrier index of the combined SREQ. The DiFuse AP extracts the SREQ of the selected user from the combined SREQ, which we call SREP-I, which is then broadcast to all remaining users. The user checks if SREP-I matches the previously sent SREQ, if so, then it transmits the actual CSI feedback.

*Example*

Let us assume that the capacity gain of User 2 and User 3 are 1.750 and 1.703, respectively. Then, with slot thresholds of  $\delta = 0.99$  and  $i = 2$ , users will send their SREQs to the AP as shown in Figure 4.5 (a). Since only User 2 is associated with the highest activated subcarrier index, the contention is successfully resolved.

- **Collision case**

A collision case happens when more than one users are associated with the highest activated subcarrier index of the combined SREQ. Since the SREQs do not include any user information such as user ID or address, the SREP-I will invoke multiple users to send their CSI feedbacks at the same time. If the AP fails to apprehend the received CSI due to the collision, it just quits the user selection procedure and starts the data transmission for the already selected users.

*Example*

Let us assume that the capacity gains of User 2 and User 3 are 1.75. Then, their SREQs will be the same and finally CSI collision will occur, as shown in Figure 4.5 (b).

- **Idle case**

An idle case happens if there are no associated users for SREQ, which means  $\lambda_{i,k} < E_i^{min}$ , for all users (Figure 4.5 (c)). Therefore, in this case, the SREQ transmissions do not occur. Instead, after timeout (2 SIFS), the AP transmits the pre-defined OFDM symbols (we call this SREP-II) to inform the users of the idle case. Compared to the collision case, the idle case hardly affects the system performance, since the transmission times of SREQ and SREP are very short (a few microseconds per each).



Based on this fact, we allow users to perform re-mapping and re-transmission of SREQs again, when an idle case happens. Specifically, in each retry, users make SREQs with different slot thresholds. Note that the number of retries is limited by the pre-defined threshold. The SREP-I and SREP-II may require some modifications to the legacy standard, but we note that this change can reduce up to 3 OFDM symbol transmissions, compared to the legacy polling frame.

*Example*

Let us assume that the capacity gains of User 2 and User 3 are 1.690 and 1.691, respectively. With slot thresholds of  $\delta = 0.99$  and  $i = 2$ , each user cannot generate SREQ because their capacity gains are too low. After receiving SREP-II, users regenerate SREQs with a decreased  $\delta$  (e.g., 0.95). Here, users can successfully generate SREQs and User 3 will be selected as the  $i$ th user, as shown in Figure 4.5 (c).

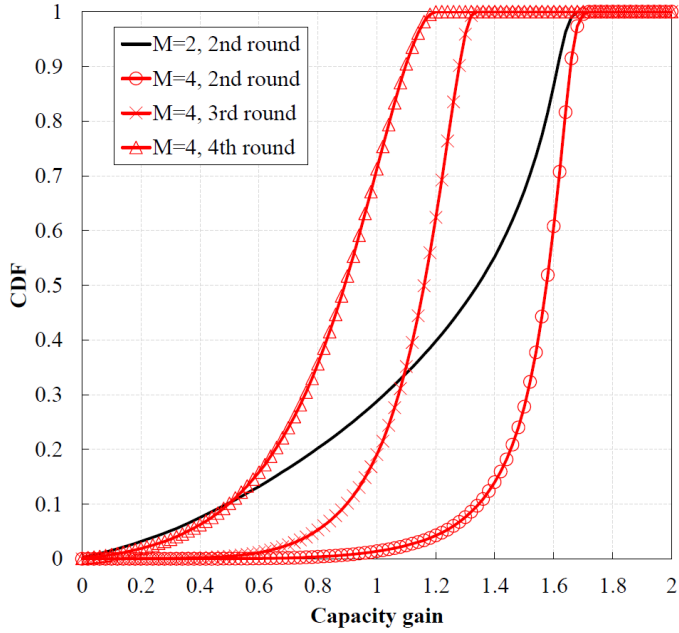


Figure 4.6: **Capacity gain distribution for  $M = 2$  and  $4$ .** As the selection round advances, the average capacity gain decreases. This result is predictable, because the power allocated to each user is reduced as a user group size increases.

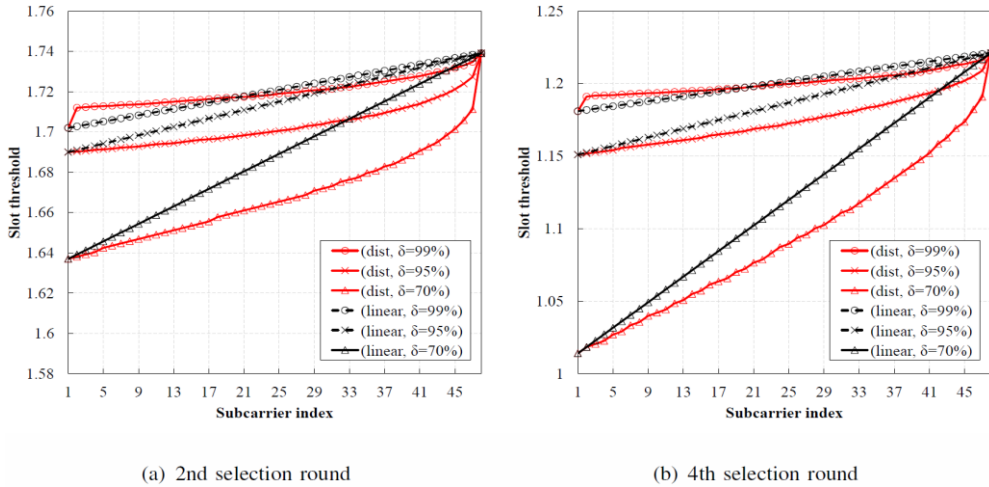


Figure 4.7: **Examples of slot thresholds for different mapping configurations.** In the ‘dist’ design, gaps between slot thresholds are determined by the capacity gain distribution, while in the ‘linear’ design, gaps are determined to be equal.

### 4.3.3 Slot Threshold Design

To maximize the number of success cases, we devise an empirical method to dynamically set the slot thresholds of DiFuse. We address several challenges to accomplish this. First, it is not easy to know the exact channel status of users before the user selection. For this reason, we set slot thresholds empirically from the channel status statistics. To do this, the AP collects the channel status of the users and computes the capacity gain distribution. We provide an example of the capacity gain distribution in Figure 4.6. Note that the capacity gain distribution is computed offline.

Second, it is difficult to reduce both collision and idle cases since we could not estimate the dynamics of user channels. Thus, we aim to avoid only collision cases, in that the penalty of the idle case is far smaller than that of the collision case. To accomplish this, we let the distance between  $E_i^{min}$  and  $E_i^{max}$  be relatively short. However, if the two values are too close together, it decreases the multi-user diversity gain because it is likely to have very few or even no associated users between the two values (i.e., idle case). On the other hand, if the two values are too far apart, it increases the probability that more than one users are associated with the same activated subcarrier (i.e., collision case).

Based on the above insight, we allow multiple slot thresholds for one selection round. In particular, we first fix  $E_i^{max}$  ( $E_i^{max}$  is set to  $F_i^{-1}(1)$ , since it reflects the expected gain that the system can obtain) and make several  $E_i^{min}$  values according to  $\delta$  (seed parameter) as follows:

$$E_i^{min} = F_i^{-1}(\delta) \quad (33)$$

, where  $F_i$  is the CDF of capacity gain distribution for selection round  $i$ .

Since the collision probability decreases with  $\delta$ , we set the initial  $\delta$  as 0.99. After that, if an idle case happens,  $\delta$  is decreased and users retry SREQ transmission with slot thresholds of the new  $\delta$ . In our case the minimum available value of  $\delta$  (denoted by  $\delta_{thr}$ ) is 70% ( $F_3^{-1}(\delta) \cong 1$  (see Figure 4.6).)

Once  $E_i^{min}$  and  $E_i^{max}$  are given, other remaining thresholds can be determined in various ways. For example, the gap between the thresholds may increase or decrease according to some functions (e.g., ‘linear’, ‘exponential’), or we can make it by

reflecting the capacity gain distribution itself (we call this ‘dist’ design). Figure 4.7 shows the examples of slot thresholds according to different mapping designs. As we will show later in Figure 4.15, the performance of ‘dist’ design is better than that of ‘linear’ design. After determining a set of slot thresholds, the DiFuse AP periodically broadcasts them to all users. Note that we do not need to broadcast slot thresholds for every packet transmission.

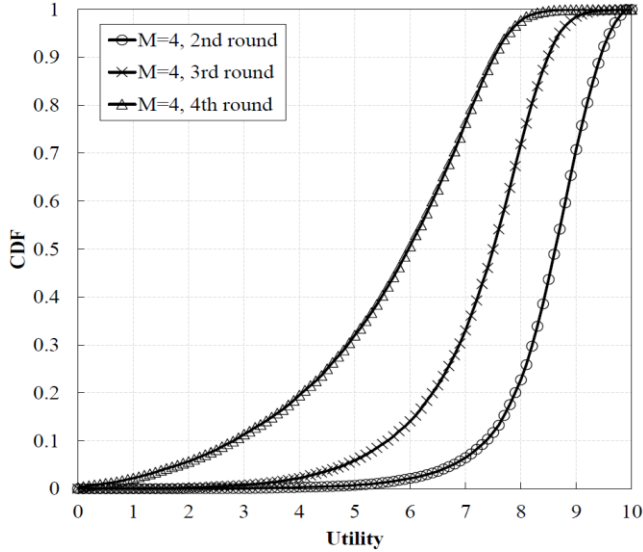


Figure 4.8: The gain distribution of the proportional fair utility for  $M = 4$ .

#### 4.3.4 Proportional Fair Selection

The fairness among users is an important factor in communication systems. We introduce DiFuse-PF which considers the proportional fairness based on DiFuse. DiFuse-PF uses the following utility to select users:

$$\frac{T_k}{\mathcal{R}_k} \quad (34)$$

, where  $T_k$ ,  $\mathcal{R}_k$  are the current available data rate and average data rate of user  $k$ , respectively.

The main difficulty of implementing the proportional fair selection lies in the fact that the AP does not know the instantaneous data rates of the users ( $T_k$ ) when the decision has to be made. However, in DiFuse-PF, each user can easily estimate  $T_k$  from the sum-capacity computation in the polling phase. Moreover, the DiFuse-PF users must use the gain of (34) to mark the subcarrier in their SREQs. Figure 4.8 shows the utility gain distribution. From the result, we see that the utility gain distribution has a similar pattern with the capacity gain distribution in Figure 4.6, but it has a different scale. The remaining procedure is essentially the same with the technique described in 4.3.3.

Table 4.1: Summary of User Selection Protocols

Protocols	Fist user selection	$i(>1)$ th user selection
SUS [9]	$\ \gamma_k\ ^2$	$\ \gamma_k\ ^2$
OPUS [12]	random	SINR
OPUS-PF [12]	$\frac{1}{\mathcal{R}_k}$	SINR
DiFuse	$\ \mathbf{h}_k\ ^2$	$\lambda_k$
DiFuse-PF	$U_k$	$U_k$

### 4.3.5 Discussions

#### 1) First User Selection

For the first user selection, we use the same technique introduced in Chapter III. We summarize the metric comparison of several user selection protocols stated so far in Table 4.1.

#### 2) Gain Reduction Compared to the Optimum

DiFuse may not give the optimal result due to the nature of the incremental selection procedure. To obtain better capacities, we may need an additional procedure to switch the existing selected users with new users, similar to GUSS [11]. However, this may lead to more interactions between the users and an AP, thus resulting in higher overhead and diminishing the achieved gain.

#### 3) Computational Complexity

In computing the sum-capacity, the complexity mainly lies on the channel inversion. For a subchannel, the complexity for channel inversion is  $O(M^3)$ , in the worst case where  $M = |S|$ . According to the recent result [58], the absolute processing time is actually affordable when  $M$  is modest ( $<50$ ). For example,  $M = |S| < 12$ , the channel inversion only takes merely  $10\mu s$ .

#### 4) Selfish User Behavior

A selfish user could manipulate the SREQ to be selected for the MU-MIMO transmission by setting a bit on the highest subcarrier index. However, the DiFuse AP

can easily detect such selfish user behavior by comparing the actual CSI feedback and the user's SREQ value.

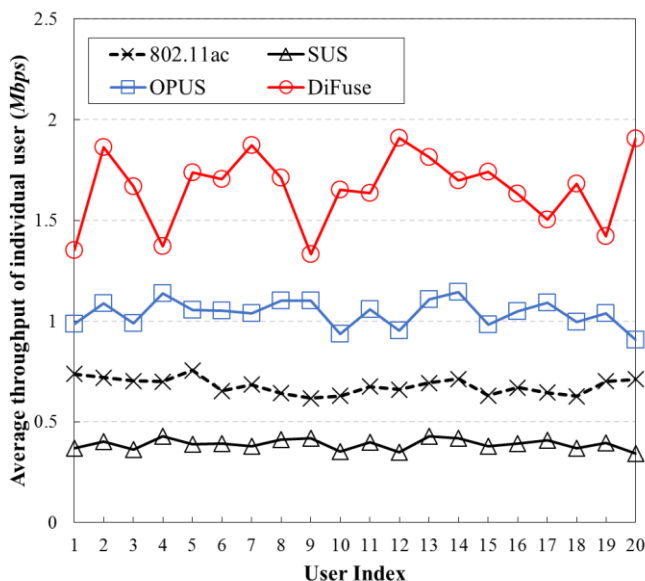


Figure 4.9: **Throughput comparison of 802.11ac, SUS, OPUS and DiFuse under hidden terminal environment.** Three users (ID 1, 2, 3) are hidden from other users.

### 5) Hidden Terminal Problem

Since DiFuse users exploit CSI feedback overhearing, in some cases, some users may not be able to participate in the selection decision due to the hidden terminal problem. Even though they cannot join the transmission group, such users may be rewarded by the proportional fair selection (DiFuse-PF). Furthermore, the CSI feedbacks are reliably transmitted at the basic data rate, so that practically, the hidden terminal rarely occurs.

Figure 4.9 shows the throughput comparison of 802.11ac, SUS, OPUS and DiFuse under hidden terminal environment. We let user ID 1 to ID 3 be hidden from other users. From the result, we can clearly observe that the hidden terminal problem shows less impact on the performance. This is because the users can be rewarded by the AP's first user selection, even if they are likely to be excluded from the selection decision due to the hidden terminal problem.



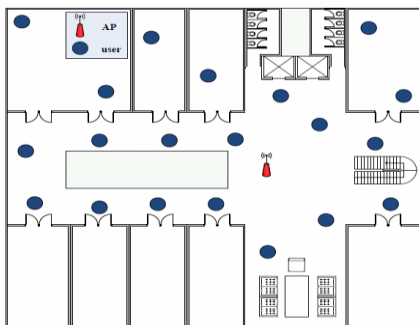


Figure 4.10: **Experiment environment.**

## 4.4 Performance Evaluation

In this section, we first evaluate the frequency domain contention scheme through the benchmark testbed. Then we evaluate the system-level performance via the trace-driven emulations.

### 4.4.1 Micro Benchmark

#### 1) *Benchmark Testbed Setup*

We implement the USRP/GNURadio testbed on OFDM PHY of 64 FFT size and 48 data subcarriers [13]. All nodes are equipped with USRP N210 on SBX daughterboard and work on 10MHz bandwidth<sup>7</sup>. The AP is equipped with  $M = 2, 4$  antennas and the user nodes are randomly located as shown in Figure 4.9. In USRP/GNURadio testbeds, real-time experiments generally have limitations; when the software radios exchange signal samples between the host and the RF front end, it incurs very high latency [38]. This limitation exacerbates under larger number of  $K$ . For this reason, we evaluate the performance of frequency domain contention under the USRP testbed, and conduct remaining parts as a trace-driven emulation approach similar to recent experimental works [6], [36], [61], [62].

---

<sup>7</sup> In trace-driven emulations, the bandwidth is set to 20MHz.

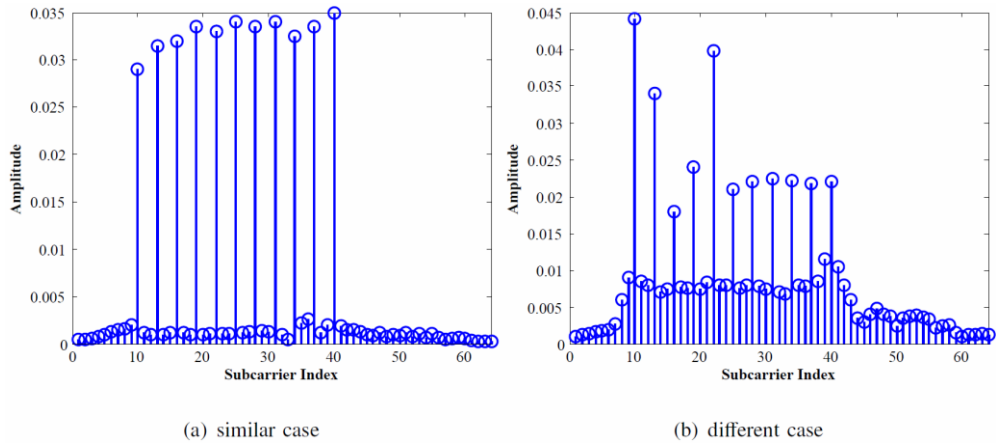


Figure 4.11: **FFT results under two different transmit power cases.**

## 2) SREQ Detection

To evaluate the feasibility of multiple users simultaneously sending the SREQ, we measure the SREQ detection probability by the AP. We let 8 users each transmit the SREQ to the 4-antenna AP under two different circumstances. For similar case, the difference between the maximum and minimum received signal power is less than 5dB, and otherwise it is different case. To control the dedicated power levels, we make gain adjustments using the transmit power control before starting the real measurements. As for SREQ, we assign different subcarrier indexes from  $\{10,13,\dots,40\}$  for each user. However, four indexes (31,34,37,40) are assigned to the last user. Also all 8 users are connected to a central controller to achieve transmission synchronization.

The FFT result examples regarding the two cases are compared in Figure 4.10. The threshold for detection is set to 10dB over noise level. As expected, the result of the similar case is clearer. In the different case, though the noise level increases due to the power difference, we can still accurately detect the signals at all 11 dedicated points.

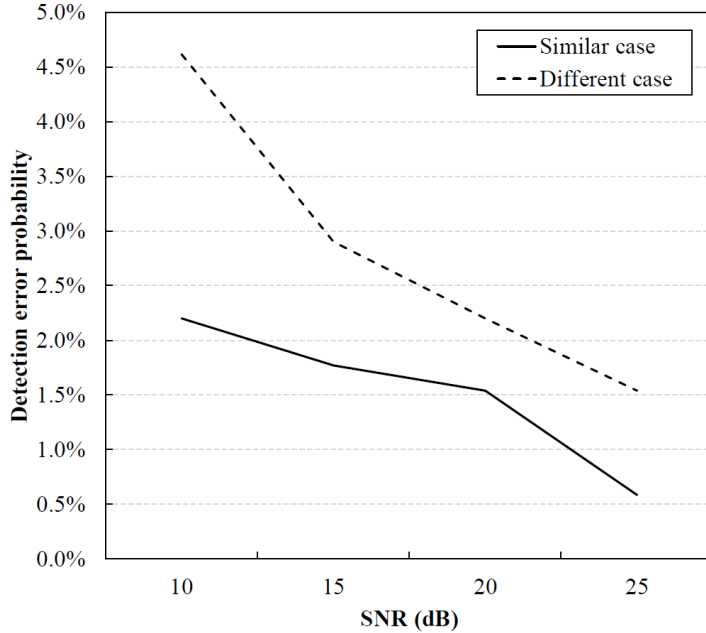


Figure 4.12: **Detection error probability vs. SNR.** The SNR is synthesized from the SNRs of multiple SREQs. Even in the worst case, the detection error of DiFuse is still less than 5%.

Next, we measure the detection error probability according to the received SNR synthesized from the multiple SREQs. We define the detection error as the event that the AP mis-detects the SREQ of the largest subcarrier index, i.e., 40. Recall that the SREQ of the largest subcarrier index, i.e., SREQ from the highest capacity gain user, matters most in DiFuse. As shown in Figure 4.11, in the similar case, the detection error is only about 0.6% at the high SNR range. The detection error increases in the different case, but is always less than 5% even in the low SNR range (the majority of detection error is due to the false negative.). The signal mis-detection in subcarrier level comes from two major sources, interference and misalignment. For example, back2F [36] suffers from the strong self-signal interference by the full-duplexing. In contrast, in DiFuse, only the noise can affect the detection performance, since control frames play a role of holding the medium, like RTS/CTS. Also, as long as the misalignment is less than CP, the AP can still detect all SREQs reliably, as mentioned earlier.

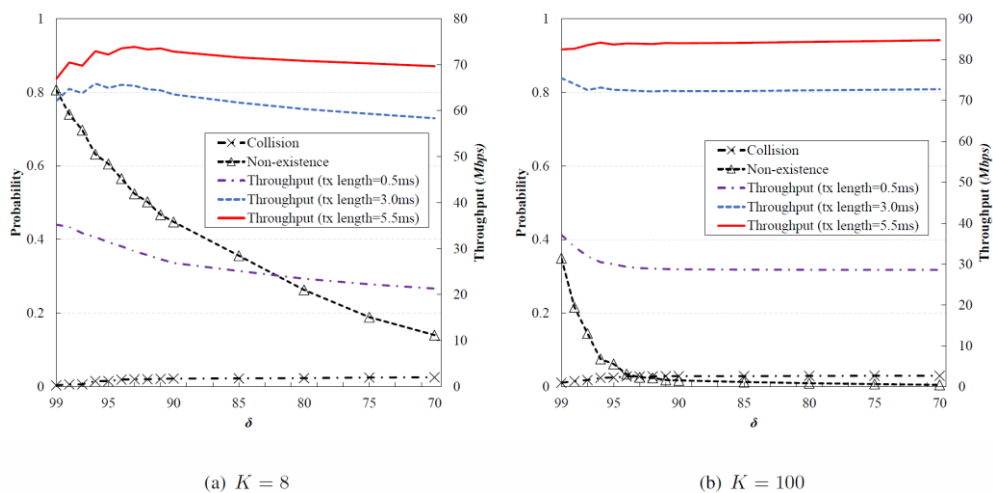


Figure 4.13: **Performance comparison on the frequency domain contention.** The DiFuse can limit the collision probability to 3% for 100 users.

### 3) Frequency Domain Contention

To evaluate the frequency domain contention of DiFuse, we measure two probabilities of collision case ( $p_{col}$ ) and idle case ( $p_{idle}$ ) according to different  $\delta_{thr}$ . Each value is averaged over whole polling rounds. Based on the results, we additionally compute the expected throughput according to different transmission lengths (0:5ms, 3:0ms, 5:5ms). For comparison, we conduct an emulation for the case of  $K = 100$ , and illustrate both results in Figure 4.12.

From the result, we can clearly observe the tradeoff between  $p_{idle}$  and  $p_{col}$ :  $p_{idle}$  increases with  $\delta_{thr}$ , while  $p_{col}$  decreases. First, the initial  $p_{idle}$  is quite high in both cases. Specifically, in the case of  $K = 8$ , the actual number of contending users is very small (i.e., 7,6,5 for each selection round, respectively), and thus  $p_{idle}$  is much higher than that of  $K = 100$ . Actually this high piddle shows that our slot threshold design fits well in the real scenario. Recall that our design mainly aims to lower  $p_{col}$ , at the expense of the increase of  $p_{idle}$ . As a result,  $p_{col}$  of two cases are 2.5% and 2.9%, respectively. And thus we can effectively limit the impact of  $p_{col}$  on the system performance (e.g., throughput).

From the viewpoint of the throughput of different  $K$ , we can see that their patterns are different from each other. For small  $K$ , the throughput decreases with  $\delta_{thr}$ . This result indicates that the throughput of small  $K$  is highly affected by the re-transmission overhead in the frequency domain contention. In particular, the small  $K$  increases  $p_{idle}$ , and thus invokes more re-transmissions of SREQs and SREPs. On the other hand, for large  $K$ ,  $p_{idle}$  converges to 0 quickly, and thus it is very unlikely to go into the re-mapping procedures. As a result, the throughput seems to be not much affected by  $\delta_{thr}$ .

In addition, we can see that there is an optimal  $\delta_{thr}$  point to maximize the throughput in each case. For small  $K$ , except the case of transmission length of 0.5ms, the throughput is slightly increased before going down. For large  $K$ , the results show the similar pattern to those of small  $K$ , but the gap is very small because retries of SREQ/SREP transmission rarely happen.

#### 4.4.2 System-Level Performance

##### 1) *Setting*

We use MATLAB to emulate 802.11ac, SUS<sup>8</sup>, OPT, OPUS, OPUS-PF, DiFuse and DiFuse-PF with the basic parameters defined in the 802.11ac specification [2].  $P$  is set to 15dB and we use the ESNR (Effective SNR) based rate adaptation scheme [62]. We use our CSI traces obtained from the USRP testbed. Our traces basically contain 10,000 per-subcarrier CSI traces for 20 users. For large  $K$  (e.g., 200), we extend the trace by multiplying random complex numbers. Unless otherwise stated,  $K, M$  and transmission length is set to 20ms, 4ms and 0.5ms, respectively.

---

<sup>8</sup> We set the parameter  $\alpha$  used in SUS as 1, and so it does not render the early termination.

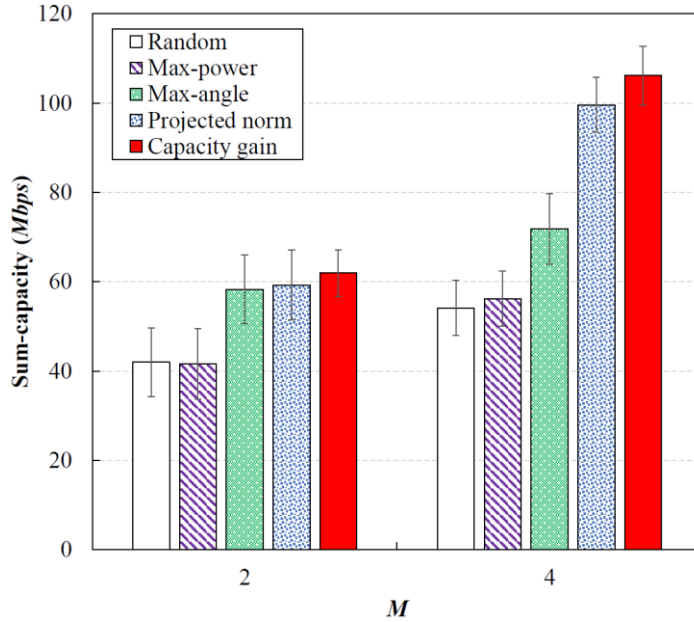


Figure 4.14: **Sum-capacity vs.  $M$** . The capacity gain metric which is used in DiFuse achieves the sum-capacity improvement of 2.0x, 1.9x, 1.6x and 1.1x over random (802.11ac), max-power, max-angle and projected norm (SUS, OPUS), respectively.

## 2) The Impact of Selection Metrics

In this section, to evaluate the impact of selection metrics, we compare the sum-capacities of five different selection metrics according to  $M$ , in Figure 4.13. The first user is randomly selected for all cases. Note that this result does not include the CSI feedback overhead.

From the results, we can observe that capacity gain metric consistently outperforms other metrics. The capacity improvements of capacity gain metric which is used in DiFuse are up to 2.0x, 1.9x, 1.6x and 1.1x compared to random, max-power, max-angle and projected norm, respectively. As expected the random user selection gives the worst performance. The performance of max-power is better than the random selection, but its gain is marginal. The orthogonality gives more capacity gains than the channel strength, but using only this metric alone limits the performance, especially when  $M$  is large. On the other hand, the projected norm metric shows a quite good performance, since it considers the tradeoff between channel strength and

orthogonality. However, as discussed in Section 4.3, this metric cannot guarantee the highest sum capacity gain in every selection round. In our result, 16 % of total cases suffer from that problem.

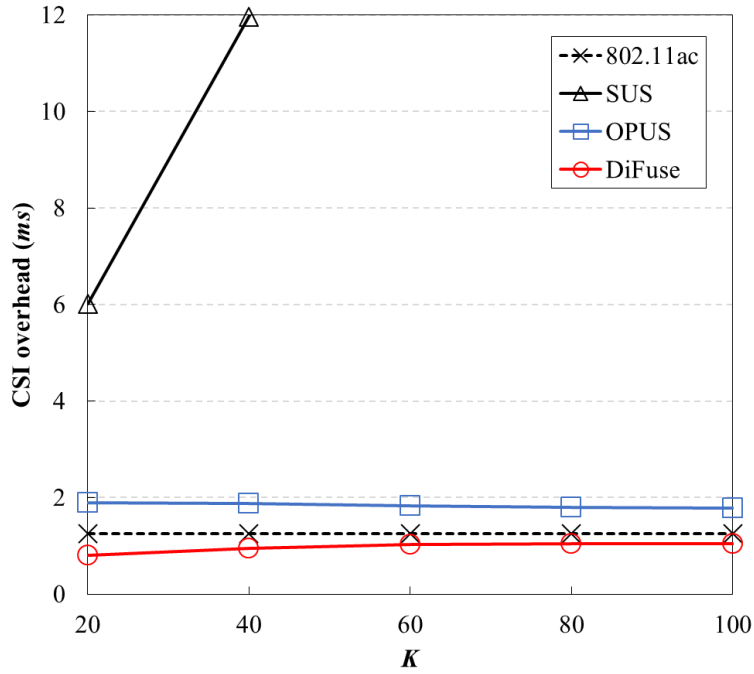


Figure 4.15: **CSI feedback duration vs.  $K$** . The average feedback time of DiFuse is lower than those of other protocols.

### 3) CSI Feedback Overhead

In this section, we investigate the CSI feedback overhead of each protocol. Figure 4.14 shows the average CSI feedback duration as a function of  $K$ . First, in 802.11ac the CSI feedback overhead is fixed regardless of  $K$  because it always gathers CSI of users as many as  $M$ . SUS conveys a much larger overhead than other schemes, constantly increasing with  $K$ . We observe that OPUS shows a higher overhead than 802.11ac. Even though OPUS limits the CSI overhead by terminating the selection earlier than 802.11ac, it suffers from the inefficient time-domain contention. In the meantime, DiFuse has the lowest overhead over all cases, thanks to the frequency domain contention and quick termination. Such overhead reduction further improves the throughput performance of DiFuse based on the sum-capacity enhancement by the user selection method.



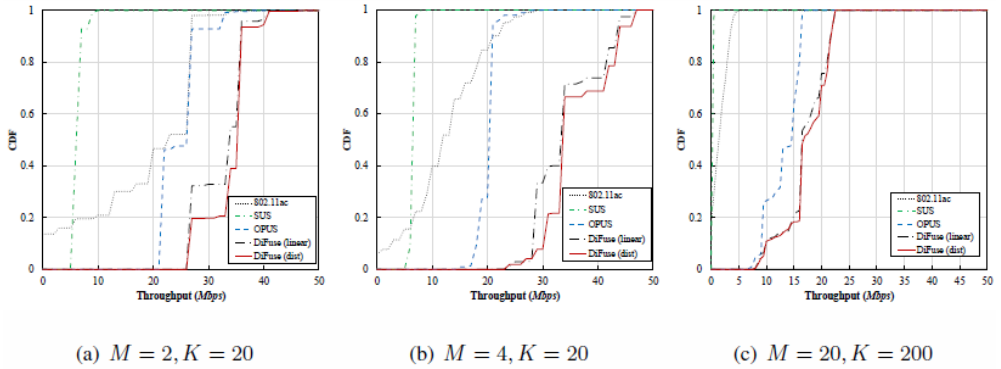


Figure 4.16: **Throughput vs.  $M$ .** DiFuse obtains the maximum throughput gain of 4.8x, 2.8x and 1.8x on average, over 802.11ac, SUS, and OPUS, respectively. In massive MIMO case (c), DiFuse still obtains modest throughput even under significantly large CSI reports.

#### 4) Throughput

We first compare the throughput of each protocol in Figure 4.15, under different network configurations. DiFuse outperforms other protocols in all cases. Except for the case of (c), DiFuse obtains the maximum throughput gain of 4.8x, 2.8x and 1.8x on average, over 802.11ac, SUS, and OPUS, respectively, especially when DiFuse uses the ‘dist’ slot thresholds design. Through the ‘dist’ design, DiFuse can achieve the additional gain of maximum 5% than the ‘linear’ design. The gap between DiFuse and OPUS becomes even larger with increasing  $M$ . First, the frequency domain contention of DiFuse gives a much smaller overhead than the contention scheme used in OPUS. Second, the effective selection method of DiFuse provides a higher capacity than that of OPUS.

Additionally, we observe zero-throughput cases in the case of 802.11ac. This is the limitation of ZFBF. The AP wastes most transmit power for interference cancellation and the intended signal may have low power. This can lead to zero-throughput. 802.11ac significantly suffers from the zero-throughput, due to the random user selection. SUS achieves better performance than 802.11ac, but the gain over 802.11ac is not that large due to its long CSI feedback time.

In massive MIMO [28], [29], [31], [58] or distributed MIMO systems [63], [64], we can exploit much more transmit antennas. Figure 4.15 (c) shows the result when  $M =$

20,  $K = 200$ . Recall that the size of each CSI report is approximately 10 times larger than that of the case (a). In 802.11ac and SUS, the performance of ZFBF drops dramatically, due to the tightness of the DoF. Recall that in both protocols, the number of selected users is  $M$ . In result, almost zero-throughput happens in both 802.11ac and SUS. In contrast, the results of DiFuse and OPUS show that they are feasible under many-antenna systems in practice. Also, DiFuse brings a higher throughput than OPUS in this scenario.

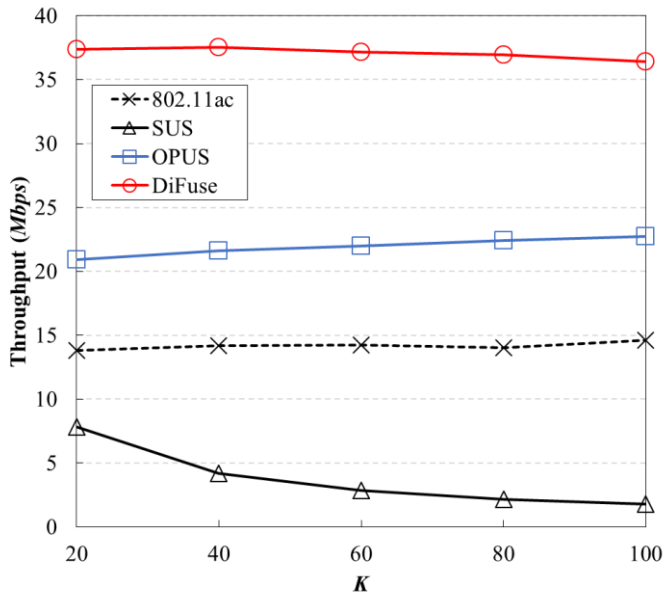


Figure 4.17: **Throughput vs.  $K$ .** DiFuse consistently outperforms other protocols. DiFuse and OPUS benefit from the multi-user diversity gain.

Next, we measure the throughput of four protocols, according to  $K$  in Figure 4.16. The random selection of 802.11ac gives constant throughput regardless of  $K$ . We can clearly see that the throughput of SUS decreases with  $K$ . Meanwhile, OPUS and DiFuse persistently obtain higher throughput over other two protocols, due to the small number of CSI feedback transmissions, plus the capacity enhancement by selection. Specifically, DiFuse outperforms OPUS over all cases.

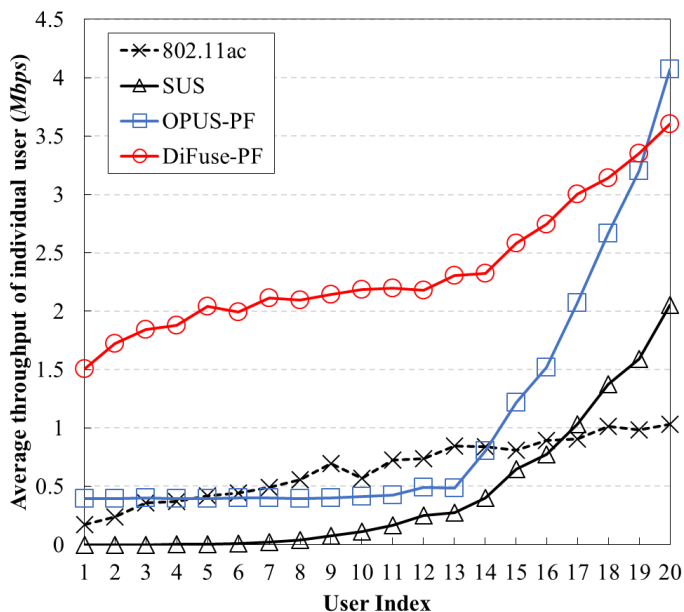


Figure 4.18: **Throughput comparison of 802.11ac, SUS, OPUS-PF and DiFuse-PF.** DiFuse-PF achieves the best fairness even when the users experience diverse channel qualities, while maintaining the throughput.

### 5) Fairness

To evaluate the fairness performance of DiFuse-PF, we let 20 users experience diverse average SNRs ranging from 5dB to 20dB; the user with a larger ID number has a higher SNR than the user with a lower ID number.

Figure 4.17 shows the throughput that each user attains under each protocol. 802.11ac severely suffers due to the low SNR users, diminishing the MU-MIMO effectiveness. Although SUS shows higher throughput than 802.11ac, some users with low SNR suffer from the starvation. OPUS-PF [12] is the proportional fair version of OPUS, where the user with the lowest average throughput is likely to be selected as the first user, so that users with low SNRs can maintain throughput. However, just considering the first user for user fairness is not enough, especially when users experience diverse channel qualities. Users with high channel gains will eventually join the MU-MIMO transmission group in the following rounds. Recall that OPUS-PF and OPUS both use the same selection metric to select users except

for the first user. Meanwhile, DiFuse-PF shows better fairness than others since the probability that users with low SNRs have a transmission chance is increased.

## **4.5 Conclusion**

In this chapter, we have proposed a new user selection protocol called DiFuse, which employs the sum-capacity gain as the user selection metric and exploits the frequency domain contention to reduce CSI feedback overhead. To evaluate the performance of DiFuse, we conduct USRP/GNURadio based experiments as well as the extensive trace-driven emulations. The results show that DiFuse consistently outperforms other schemes in terms of throughput and proportional fairness.

# CHAPTER V

## Conclusion

This dissertation proposes three co-designs of PHY/MAC layers for MIMO based Wi-Fi networks, in order to overcome the limitations of current MIMO based Wi-Fi networks and also improve the network capacity. We first show that 802.11mc, collision resolution based MU-MIMO MAC protocol for heterogeneous MIMO based Wi-Fi systems can effectively resolves RTS frame collision and extracts CSI from the resolved RTS frames for simultaneous data transmissions. Next, we propose a practical user selection scheme for MU-MIMO Wi-Fi networks, called 802.11ac+. In 802.11ac+, both the AP and users participate in making a scheduling decision via channel hint exchange. As a result, the user selection in 802.11ac+ boosts the network capacity significantly. Finally, from DiFuse, we show that user selection scheme could be enhanced by using capacity gain as a selection metric and exploiting the frequency domain signaling scheme. In particular, the new metric of DiFuse enables to accomplish a higher level of network capacity and its frequency domain CSI feedback contention further reduces a feedback overhead compared to 802.11ac+ based on the time-domain CSI feedback contention.

# BIBLIOGRAPHY

- [1] *IEEE Standard for Information Technology – Local and Metropolitan Area Networks – Specific Requirements-Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput*, IEEE Std. 802.11n, 2009.
- [2] *IEEE Draft Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks*, IEEE Std. 802.11ac\_D5.0, 2013.
- [3] *Evolved Universal Terrestrial Radio Access (E-UTRA), Downlink Multiple Input Multiple Output (MIMO) enhancement for LTE-Advanced (Release 11)*, 3GPP TR V11.0.0.
- [4] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, “Performance Anomaly of 802.11b,” in *Proc. IEEE INFOCOM*, 2003.
- [5] K. C. J. Lin, S. Gollakota, and D. Katabi, “Random Access Heterogeneous MIMO Networks,” in *Proc. ACM SIGCOMM*, 2011.
- [6] X. Xie, X. Zhang, and K. Sundaresan, “Adaptive Feedback Compression for MIMO Networks,” in *Proc. ACM MOBICOM*, 2013.
- [7] G. Dimic and N. Sidiropoulos, “On Downlink Beamforming with Greedy User Selection: Performance Analysis and a Simple New Algorithm,” *IEEE Trans. Signal Processing*, 53(10):3857-3868, 2005.

- [8] T. Yoo and A. Goldsmith, "Sum-Rate Optimal Multi-Antenna Downlink Beamforming Strategy based on Clique Search," in *Proc. IEEE GLOBECOM*, 2005.
- [9] T. Yoo and A. Goldsmith, "On the Optimality of Multiantenna Broadcast Scheduling using Zero-Forcing Beamforming," *IEEE Journal on Selected Areas in Communications*, 24(3):528-541, 2006.
- [10] L. Jin, X. Gu and Z. Hu, "Low-Complexity Scheduling Strategy for Wireless Multiuser Multiple-Input Multiple-Output Downlink System," *IET Communications*, 5(7):990-995, 2011.
- [11] S. Huang, H. Yin, J. Wu, VCM. Leung, "User Selection for Multi-User MIMO Downlink with Zero-Forcing Beamforming," *IEEE Trans. Vehicular Technology*, 62(7):3084-3097, 2012.
- [12] X. Xie and X. Zhang, "Scalable User Selection for MU-MIMO Networks," in *Proc. IEEE INFOCOM*, 2014.
- [13] Ettus INC., Universal Software Radio Peripheral. [Online]. Available: <http://ettus.com>
- [14] K.C.-J. Lin, N.Kushman, and D.Katabi, "ZipTx: Harnessing Partial Packets in 802.11 Networks," in *Proc. ACM MOBICOM*, 2008.
- [15] B. Han, A. Schulman, F. Cringoli, N. Spring, B. Bhattacharjee, L. Nava, L. Ji, S. Lee and R. Miller, "Maranello: Practical Partial Packet Recovery for 802.11," in *Proc. USENIX NSDI*, 2010.
- [16] S. Gollakota and D. Katabi, "ZigZag Decoding: Combating Hidden Terminals in Wireless Networks," in *Proc. ACM SIGCOMM*, 2008.
- [17] X. Zhang and K. G. Shin, "Chorus: Collision Resolution for Efficient Wireless Broadcast," in *Proc. IEEE INFOCOM*, 2010.
- [18] S. Katti, S. Gollakota, and D. Katabi, "Embracing Wireless Interference: Analog Network Coding," in *Proc. ACM SIGCOMM*, 2007.



- [19] J. Peng, L. Cheng, and B. Sikdar, "A Wireless MAC Protocol with Collision Detection," *IEEE Trans. Mobile Computing*, 6(12):1357–1369, Dec. 2007.
- [20] S. Sen, R. R. Choudhury, and S. Nelakuditi, "CSMA/CN: Carrier Sense Multiple Access with Collision Notification," *IEEE Trans. Networking*, 20(2):544–556, Apr. 2012.
- [21] K. Tan, H. Liu, J. Fang, W. Wang, J. Zhang, M. Chen and G. M. Voelker, "SAM: Enabling Practical Spatial Multiple Access in Wireless LAN," in *Proc. ACM MOBICOM*, 2009.
- [22] W. L. Shen, Y. C. Tung, K. C. Lee and K. C.-J. Lin, "Rate Adaptation for 802.11 Multiuser MIMO Networks," in *Proc. ACM MOBICOM*, 2012.
- [23] S. Gollakota, S. D. Perli, and D. Katabi, "Interference Alignment and Cancellation," in *Proc. ACM SIGCOMM*, 2009.
- [24] K. Lin, Y. J. Chuang, and D. Katabi, "A Light-Weight Wireless Handshake," *ACM SIGCOMM Comput. Commun. Rev.*, pp. 28–34, 2012.
- [25] G. J. Foschini, "Layered Space–Time Architecture for Wireless Communication in a Fading Environment when Using Multi-Element Antennas," *Bell Labs Techn. J.*, 1(2):41–59, 1996.
- [26] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [27] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [28] C. Shepard, H. Yu, N. Anand and L. E. Li, "Argos: Practical Many-Antenna Base Stations," in *Proc. ACM MOBICOM*, 2012.
- [29] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of Cellular Networks: How Many Antennas do We Need?" *IEEE Journal on Selected Areas in Communications*, 31(2):160–171, Feb. 2013.

- [30] H. Huh, G. Caire, H. C. Papadopoulos, and S. A. Ramprasad, "Achieving Large Spectral Efficiency with TDD and Not-So-Many Base-Station Antennas," in *Proc. IEEE-APS Topical Conf. APWC*, 2011.
- [31] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors and F. Tufvesson, "Scaling Up MIMO: Opportunities and Challenges with Very Large Arrays," *IEEE Signal Process. Mag.*, 30(1): 40–60, Jan. 2013.
- [32] S. H. Y. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust Rate Adaptation for 802.11 Wireless Networks," in *Proc. ACM MOBICOM*, 2006.
- [33] G. Holland, N. Vaidya, and P. Bahl, "A Rate-Adaptive MAC protocol for Multi-Hop Wireless Networks," in *Proc. ACM MOBICOM*, 2001.
- [34] J. Choi, J. Yoo, S. Choi, and C. Kim, "EBA: An Enhancement of the IEEE 802.11 DCF via Distributed Reservation," *IEEE Trans. Mobile Computing*, 4(4):378–390, Aug. 2005.
- [35] G. Bianchi and I. Tinnirello, "Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 network," in *Proc. IEEE INFOCOM*, 2003.
- [36] S. Sen, R. R. Choudhury, and S. Nelakuditi, "No Time to Countdown: Migrating Backoff to the Frequency Domain," in *Proc. ACM MOBICOM*, 2011.
- [37] J. Jeong, S. Choi, J. Yoo, S. Lee, and C.-K. Kim, "Physical Layer Capture aware MAC for WLANs," *Wireless Netw.*, 19(4):533–546, May 2013.
- [38] T. Schmid, O. Sekkat, and M. B. Srivastava, "An Experimental Study of Network Performance Impact of Increased Latency in Software Defined Radios," in *Proc. 2nd ACM Int. Workshop Wireless Netw. Testbeds, Exp. Eval. Characterization*, 2007.
- [39] MadWiFi. [Online]. Available: <http://madwifi-project.org>
- [40] Z. Shen, R. Chen, J. Andrews, R. Heath and B. Evans, "Low Complexity User Selection Algorithms for Multiuser MIMO Systems with Block

- Diagonalization,” *IEEE Trans. Signal Processing*, 54(9):3658-3663, Sep, 2006.
- [41] Q. H. Spencer, A. L. Swindlehurst and M. Haardt, “Zero-forcing Methods for Downlink Spatial Multiplexing in Multiuser MIMO Channels,” *IEEE Trans. Signal Processing*, 52(2):461-471, 2004.
- [42] C. Hellings, M. Joham and W. Utschick, “Gradient-Based Rate Balancing for MIMO Broadcast Channels with Linear Precoding,” in *Proc. IEEE Smart Antennas (WSA), International ITG Workshop on*, 2011.
- [43] C. Guthy and W. Utschick, “Sum Throughput Maximization in Quality of Service Constraint Multiuser MIMO Systems based on Perturbation Analysis,” in *Proc. IEEE Smart Antennas (WSA), International ITG Workshop on*, 2011.
- [44] F. Lima, M. T. Ferreira, W. Freitas and C. F. Porto, “Improved Spectral Efficiency With Acceptable Service Provision in Multiuser MIMO Scenarios,” *IEEE Trans. Vehicular Technology*, 63(6):2697-2711, 2014.
- [45] D. Love, R. Heath, W. Santipach and M. Honig, “What is the Value of Limited Feedback for MIMO Channels?,” *IEEE Communications Magazine*, 42(10):54-59, 2004.
- [46] P. Frank, A. Muller and J. Speidel, “Fair performance comparison between CQI- and CSI-based MU-MIMO for the LTE downlink,” in *Proc. Wireless Conference (EW), European*, 2010.
- [47] K. Huang, R. Heath, and J. Andrews J, “Limited Feedback Beamforming Over Temporally-Correlated Channels,” *IEEE Trans. Signal Processing*, 57(5):1959-1975, 2009.
- [48] V. Pohl, P. Nguyen, V. Jungnickel and C. H. Von, “How Often Channel Estimation is Needed in MIMO Systems,” in *Proc. IEEE Global Telecommunications Conference*, 2003.
- [49] W. Wang, A. Harada and H. Kayama, “Enhanced Limited Feedback Schemes for DL MU-MIMO ZF Precoding,” *IEEE Trans. Wireless Communications*, 12(4):1554-1561, 2013.

- [50] J. H. Lee and W. Choi, "Optimal Feedback Rate Sharing Strategy in Zero-Forcing MIMO Broadcast Channels," *IEEE Trans. Wireless Communications*, 12(6):3000-3011, 2013.
- [51] D. Love, R. Heath, VKN. Lau, D. Gesbert, B. Rao and M. Andrews, "An Overview of Limited Feedback in Wireless Communication Systems," *IEEE Journal on Selected Areas in Communications*, 26(8):1341-1365, 2008.
- [52] E. Candes and M. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, 25(2):21-30, 2008.
- [53] S. Qaseem, T. Y. AlNaouri and S. Alghadhban, "Compressive Sensing for Feedback Reduction in MIMO Broadcast Channels," in *Proc. IEEE Telecommunications (ICT)*, 2010.
- [54] D. Gesbert and M. S. Alouini, "Selective Multi-User Diversity," in *Proc. IEEE Signal Processing and Information Technology*, 2003.
- [55] D. Gesbert, M. S. Alouini, "How Much Feedback is Multi-User Diversity Really Worth?," in *Proc. IEEE ICC*, 2004.
- [56] V. Hassel, D. Gesbert, M. S. Alouini and G. Oien, "A Threshold-Based Channel State Feedback Algorithm for Modern Cellular Systems," *IEEE Trans. Wireless Communications*, 6(7):2422-2426, 2007.
- [57] W. Zhang and K. Letaief, "MIMO Broadcast Scheduling with Limited Feedback," *IEEE Journal on Selected Areas in Communications*, 25(7):1457-1467, 2007.
- [58] Q. Yang, X. Li, H. Yao, J. Fang, K. Tan, W. Hu, J. Zhang and Y. Zhang, "BigStation: Enabling Scalable Real-time Signal Processing in Large Mu-MIMO Systems," in *Proc. of ACM SIGCOMM*, 2013.
- [59] A. Bhartia, Y. C. Chen, S. Rallapalli and L. Qiu, "Harnessing Frequency Diversity in Wi-Fi Networks," in *Proc. ACM MOBICOM*, 2011.

- [60] J. Fang, K. Tan, Y. Zhang, S. Chen, L. Shi, J. Zhang, Y. Zhang, and Z. Tan, "Fine-Grained Channel Access in Wireless LAN," *IEEE Trans. Networking*, 21(3): 772–787, June 2013.
- [61] S. Lee and C.-K. Kim, "D-Fi: A Diversity-Aware Wi-Fi Using an OFDM-based Bloom Filter," in *Proc. of IEEE ICNP*, 2012.
- [62] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 Packet Delivery from Wireless Channel Measurements," in *Proc. ACM SIGCOMM*, 2010.
- [63] X. Zhang, K. Sundaresan, M. A. A. Khojastepour, S. Rangarajan, and K. G. Shin, "NEMOx: Scalable Nnetwork MIMO for Wireless Networks," in *Proc. ACM MOBICOM*, 2013.
- [64] H. S. Rahul, S. Kumar, and D. Katabi, "JMB: Scaling Wireless Capacity with User Demands," in *Proc. ACM SIGCOMM*, 2012.

초 록

# 충돌해결과 유저선택을 이용한 Wi-Fi 무선망에서의 MIMO 성능 향상 기법

이규행

전기·컴퓨터공학부

서울대학교 대학원

다수의 안테나를 이용하여 무선신호를 송수신하는 MIMO 기술은 추가적인 주파수자원 확보 없이 무선망 용량을 획기적으로 늘려줄 수 있기 때문에 차세대 무선망의 필수 핵심기술로 각광받고 있다. MIMO 는 동시에 다수의 사용자에게 전송을 하거나, 특정 유저에게 빔포밍을 하여 데이터 전송 속도를 증가시킬 수 있다. 이러한 이점으로 셀룰라망 뿐만 아니라 Wi-Fi 무선망에서도 MIMO 를 지원하는 표준들이 개발되어 왔고, 2009 년을 기점으로 상용화되기 시작하였다. 하지만, 대부분의 MIMO 기반 Wi-Fi 시스템들은 MIMO 본연의 장점을 충분히 살리고 있지 못하고 다양한 한계점을 드러내고 있다. 먼저 MIMO 노드는 기존의 SISO 노드와 채널을 공유할 경우 마치 성능 비정상 현상(Performance Anomaly)처럼 SISO 노드 수준으로 성능이 떨어지게 된다. 또한, MU-MIMO 를 위한 CSI 획득 과정은 매우 큰 MAC 오버헤드를 수반하여 전송효율을 저하시키고 더 나아가 최적 유저 선택 기법을 불가능하게 하여 전송용량을

제한시킨다. 이러한 한계점들이 발생하게 된 근본적인 원인은 현재 MIMO 시스템의 PHY 와 MAC 이 유기적으로 설계되어 있지 않기 때문이다. 즉, MIMO 의 PHY 기술이 빠르게 발전하면서 기능이 다양해지고 고도화되고 있지만, MAC 계층에서는 이러한 잠재적인 기능들을 효과적으로 이끌어내지 못하고 있으며, 단지 데이터 전송 속도 선택의 폭을 조금씩 넓혀주는 단순한 기능확장에만 초점을 맞춰왔다. 그 결과 새롭고 강력한 PHY (MIMO) 기술이 개발되어도 충분히 활용될 수 없었으며, 기존 시스템과의 호환성 문제를 일으켜 전체적인 시스템 성능이 저하되기도 하였다.

본 논문은 MIMO Wi-Fi 무선망을 위한 세가지 PHY/MAC 통합설계 기법에 대해 논한다. 먼저 포스트앰블 기반의 IAC 를 활용하여 프레임 충돌문제를 해결하고, 이기종망에서 동시전송 기회를 늘려주는 MAC 프로토콜 802.11mc 를 소개한다. 두번째로, CSI 획득 오버헤드를 획기적으로 줄이면서 실제로 구현가능한 최적 유저 선택 프로토콜 802.11ac+를 제안한다. 마지막으로, 주파수 도메인 시그널링과 더 나은 유저 선택 매트릭을 사용하여 네트워크 용량을 증대시키고 기존에 제안했던 802.11ac+의 CSI 획득 오버헤드를 한 단계 더 높은 수준으로 줄일 수 있는 DiFuse 를 제안한다. 본 논문에서는 정교한 분석과 시뮬레이션은 물론 USRP 기반 테스트베드 구현 및 실험을 통해 성능평가의 신뢰성을 높이고 제안한 프로토콜들의 구현 가능성을 입증한다.

**주요어:** 멀티유저 마이모, 매체 접근 제어, 프레임 충돌, 채널정보, 유저 선택

**학번:** 2009-23130